客服热线 400 - 820 - 9595

绵密网络 专业服务

中达电通已建立了 70 余个分支机构及服务网点，并塑建训练有素的专业团队，提供客户最满意的服务，公司技术人员能在 2 小时内回应您的问题，并在 48 小时内提供所需服务。

沈阳 电话 :(024)2334-1160	济南 电话 :(0531)8690-6277	杭州 电话 :(0571)8882-0610	重庆 电话 :(023)8806-0306	乌鲁木齐 电话 :(0991)4678-141
哈尔滨 电话 :(0451)5366-5568	太原 电话 :(0351)4039-485	合肥 电话 :(0551)6281-6777	昆明 电话 :(0871)6313-7362	兰州 电话 :(0931)6406-725
长春 电话 :(0431)8892-5060	郑州 电话 :(0371)6384-2772	武汉 电话 :(027)8544-8475	广州 电话 :(020)3879-2175	西安 电话 :(029)8836-0780
呼和浩特 电话 :(0471)6297-808	石家庄 电话 :(0311)8666-7337	南昌 电话 :(0791)8625-5010	厦门 电话 :(0592)5313-601	贵阳 电话 :(0851)8690-1374
北京 电话 :(010)8225-3225	上海 电话 :(021)6301-2827	成都 电话 :(028)8434-2075	南宁 电话 :(0771)2621-501	福州 电话 :(0591)8755-1305
天津 电话 :(022)2301-5082	南京 电话 :(025)8334-6585	长沙 电话 :(0731)8549-9156		



创 变 智 造 新 未 来

DIALink 软件使用手册

台达 DELTA
中达电通股份有限公司

地址：上海市浦东新区民夏路238号
邮编：201209
电话：（021）5863-5678
传真：（021）5863-0003
网址：http://www.delta-china.com.cn

扫一扫，关注官方微信

中达电通公司版权所有
如有改动,恕不另行通知

DIAL-494AD10-05
2023/01/11

www.deltaww.com

DELTA

DIALINK 软件使用手册

版本修订一览表

版本	变更内容	发行日期
第一版	第一版发行	2018/09/30
第二版	1.所有章节增加使用于 PLC 相关内容并将”DIALink CNC 软件使用手册”改为”DIALink 软件使用手册” 2.第一章删除原第 1.2 节-第 1.4 节内容，以全新内容取代（第 1.2 节 DIALink 机种说明、第 1.3 节软件版 DIALink、第 1.4 节硬件版 DIALink - Intel Core-i 系列主机硬件规格、第 1.5 节硬件版 DIALink - Intel Celeron N3350 系列主机硬件规格） 3.第二章章节内容及顺序重整；设备总览（刀具管理、维护管理、加工程序、参数补偿及历史警报）新增页面功能简述；新增设备及点位设定内容新增 PLC、OPCDA、Edge Computing 设备说明；触发点位更新触发条件说明；排程触发及事件与动作新增 LINE、WeChat 及 WriteTag 功能说明；追加查询功能说明；更新服务状态画面内容；系统设定更新一般设定、通知设定说明及新增排班管理、LINE 设定、WeChat 设定、Modbus Slave 设定说明 4.新增附录 A Modbus Slave 功能 5.新增附录 B IFTTT Line 及微信公众号设定 6.新增附录 C 设备点位导出汇入功能 7.新增附录 D 点位计算功能数值有效时间设定	2020/03/03
第三版	1.第 1 章 1.3 新增软件试用机制; 1.2 节更新 DIALink 机种说明 2.第 2 章 2.2.1 新增稼动率说明 ;2.2.2 新增部份设备支持 COM Port 通讯、支持 Delta PC 设备 HTTP 通讯、设备列表新增子页面按键、OPC UA 联机支持设定及节点内容值读写支持功能 ;2.2.3 新增 Edge Computing 类型 counter、timer / 运算符 arithmetic ;2.4 事件动作 :条件选项可多选移动及删除、Email 可同时通知多人 ; 2.8.1 修改第三方数据串接设定 3.新增附录 E OPC UA Server 设定 4.新增附录 F合并 Dongle Key 操作指引 5.新增附录 G SECS/GEM User Guide	2021/04/02
第四版	1.第一章 1.3.1 完全安装流程内容修订，并新增说明内容;产品型号与规格部分新增简中版的操作语系 2.第二章 2.2 设备总览内容修订 ·2.2.2 新增设备支持设备表之 PLC 项目内容修订，页面截图更新。2.2.3 新增点位各页面截图更新，2.2.5 刀具管理，增加刀具警报内容及刀具使用归零内容 2.2.6，内容及页面更新。2.2.7 新增内文，2.2.8 新增内文 ·2.2.10 新增内文 ·2.4 内文修订及页面更新 ·2.8.1	2022/01/28

版本	变更内容	发行日期
	一般设定，内容大幅增修，页面更新。 3.第三章 3.3 增加「全域授权」说明。 4.第四章 4.2 增加「资安凭证」说明。 5.第五章增加 5.1.8 及 5.1.9 及 5.1.10。 6.附录 G 在 G.1 增加说明文字，G.2 安装流程说明修订。G.2 中「使用 SECS 功能的流程顺序」。G.3.2.1【注】内容说明修订。G.3.2.3 增加说明文字。增加 G.3.8。 7.新增附录 H。 8.附录 C 画面更新	
第五版	1.第一章 产品介绍： ● 1.2 DIALink 机种说明 :修改 OPCUA 模块编码说明图。 ● 1.3 软件版 DIALink：修改操作系统表格文字。 ● 1.3.1 完全安装 DIALink：HTTPS 与 Authorization 设定画面图更换，文字修改。 ● 增加章节 1.3.3 临时软件授权。 2.第二章 ● 2.2.1 CNC 信息看板：修正稼动率说明。 ● 2.2.6 维护管理：新增维护信息图文修改。 ● 原 Cloud IoT、Webhook、MQTT、PC 摘录至附录 I。 ● 2.8.1 一般设定：增加 OPCUA 图文。 ● 2.8.1 一般设定：增加全局授权文、图更新。 3.第三章 ● 3.3 范例程序：启用全局授权说明修改。 4.第四章 ● 4.2 规格手册：增加最后一段备注说明。 5.第五章 ● 5.1.8 数据库依旧无法正常运作：修改图文。 ● 5.1.9 硬件 KEY 异常，仅能使用试用版：增、修图文。 ● 5.1.10 无法联机到 CNC 设备：修改第二段说明。 6.附录 C ● C-2 到 C-5 修改内容、增加点位说明 7.附录 H ● H1.2 启用 HTTPS：最后三张图片依照繁体中文版附图修改上次简体中文版及英文版图片。 8.增加附录 I：DIALink 第三方数据串接设定。	2023/01/11

DIALink 软件使用手册

目录

第 1 章 产品介绍

1.1 概述 1-2

1.2 DIALink 机种说明 1-2

1.3 软件版 DIALink 1-4

1.3.1 完全安装 DIALink 1-6

1.3.2 更新 DIALink 1-17

1.3.3 临时软件授权 1-19

1.4 硬件版 DIALink - Intel Core-i 系列主机硬件规格 1-20

1.4.1 外观尺寸图 1-21

1.4.2 I / O 接口示意图 1-22

1.4.3 COM 引脚定义（DB9） 1-23

1.5 硬件版 DIALink - Intel Celeron N3350 系列主机硬件规格 1-23

1.5.1 外观尺寸图 1-24

1.5.2 I / O 接口示意图 1-25

1.5.3 COM 引脚定义（DB9） 1-26

1.5.4 设定 COM 类型 1-27

1.5.5 DIO 引脚定义（DB9） 1-28

1.5.6 设定 DIO 状态 1-30

第 2 章 开始使用 DIALink

2.1 登入系统 2-3

2.2 设备总览 2-4

2.2.1 CNC 信息看板 2-6

2.2.2 新增设备 2-8

2.2.3 点位设定 2-17

2.2.4 触发点位 2-30

2.2.5 刀具管理 2-33

2.2.6 维护管理 2-36

2.2.7 加工程序 2-38

2.2.8 参数补偿 2-39
2.2.9 历史警报 2-39
2.2.10 设备设定 2-40
2.3 排程触发 2-41
2.4 事件与动作 2-42
2.5 查询功能 2-49
2.6 警报清单 2-49
2.7 服务状态 2-50
2.8 系统设定 2-51
2.8.1 一般设定 2-52
2.8.2 通知设定 2-53
2.8.3 网络设定 2-54
2.8.4 排班管理 2-54
2.8.5 Line 设定 2-56
2.8.6 WeChat 设定 2-56
2.8.7 E-mail 设定 2-57
2.8.8 MODBUS Slave 设定 2-58

第 3 章 Web API 联机
3.1 概述 3-2
3.2 规格手册 3-3
3.3 范例程序 3-6

第 4 章 MQTT 联机
4.1 概述 4-2
4.2 规格手册 4-2
4.3 程序范例 4-5

第 5 章 故障排除
5.1 软件故障排除 5-2
5.1.1 无法登入 DIALink 5-2
5.1.2 MQTT 服务状态显示 Not Started 5-5
5.1.3 Data Collector 服务状态显示 Not Started 5-7

5.1.4 Data Aggregation 服务状态显示 Not Started 5-11
5.1.5 Management 服务状态显示 Not Started 5-13
5.1.6 Notification 服务状态显示 Not Started 5-16
5.1.7 Database 服务状态显示 Not Started 5-18
5.1.8 数据库依旧无法正常运作 5-20
5.1.9 插上硬件 KEY 却无法看到正确授权，仅能使用试用版 5-26
5.1.10 无法联机到 CNC 设备 5-28

附录 A MODBUS Slave 功能
A.1 概述 A-2
A.2 规格说明 A-3

附录 B IFTTT Line 及微信公众号设定
B.1 IFTTT Line 设定 B-2
B.2 微信公众号申请教学 B-11

附录 C 设备点位导出导入功能
C.1 概述 C-2
C.2 点位导出说明 C-2
C.3 点位导入说明 C-3

附录 D 点位计算功能数值有效时间设定
D.1 概述 D-2

附录 E OPC UA Server 设定
E.1 需求及环境架构说明 E-2
E.2 OPC UA Server 服务及设定 E-2
E.2.1 OPC UA Server 服务状态 E-2
E.2.2 系统设定 E-3
E.2.3 新增设备及点位 E-4
E.3 UAExpert 安装 E-8

附录 F 合并 Dongle Key 操作指引
F.1 需求说明 F-2

F.2 合并工具（DIALinkCombinDongleKeysForm）操作说明 F-2
F.2.1 扫描目前计算机 Dongle Key 信息 F-2
F.2.2 合并 Dongle Key F-4

附录 G SECS/GEM 用户指南
G.1 架构说明 G-2
G.2 安装及操作流程表 G-2
G.3 SECS/GEM 设定说明 G-4
G.3.1 新增 SECS/GEM 设备 G-4
G.3.2 SECS/GEM 设定 G-5
G.3.2.1 SECS/GEM Application G-5
G.3.2.2 PLC 连线参数设定 G-6
G.3.2.3 I/O Mapping Function Block G-7
G.3.2.4 Alarm Manager G-8
G.3.2.5 Event Manager G-9
G.3.2.6 SV DATA G-10
G.3.2.7 DV DATA G-11
G.3.2.8 EC DATA G-12
G.3.2.9 Link V Setting G-13
G.3.2.10 Command Parameter G-14
G.3.2.11 Remote Command G-15
G.3.2.12 Spool Allow G-17
G.3.2.13 Parameter Manager G-18
G.3.2.14 PPID Mapping Manager G-19
G.3.2.15 Formatted Process Parameter G-20
G.3.2.16 Formatted Command Code G-21
G.3.2.17 Unformatted Process Parameter G-22
G.3.3 联机 G-23
G.3.4 事件 G-25
G.3.5 资讯 G-26
G.3.6 Recipe G-26
G.3.7 Operation G-27
G.3.8 修改 LOG 的进阶设定 G-27
G.4 错误排除 G-28

附录 H 全域授权与 HTTPS 设置
H.1 设定说明 H-2
H.1.1 启用全域授权差异 H-2

H.1.2 启用 HTTPS H-5

附录 I DIALink 第三方数据串接设置

I.1 第三方数据传输概述 I-3
I.2 Cloud IoT I-4
I.2.1 DIALink 的微软 Azure IoT Hub 连接设置说明 I-6
I.2.1.1 DIALink 与 Azure IoT Hub 数据传输架构 I-6
I.2.1.2 Azure IoT Hub D2C (Device to Cloud) 连接设置说明 I-6
I.2.1.3 C2D (Cloud to Device) 应用说明 I-9
I.2.2 AWS IoT 应用说明 I-12
I.2.2.1 DIALink 与 AWS IoT 数据传输架构 I-13
I.2.2.2 AWS IoT Core 连接设置说明 I-13
I.2.2.3 创建策略 I-14
I.2.2.4 创建物品 I-15
I.2.2.5 证书和密钥 I-18
I.2.2.6 DIALink AWS IoT 设置 I-20
I.2.2.7 AWS IoT 客户端检视 MQTT 信息 I-22
I.3 Webhook I-22
I.4 MQTT I-23
I.5 PC I-24
I.5.1 需求与架构说明 I-24
I.5.2 新增 PC 设备与点位 I-25
I.5.2.1 新增设备 I-25
I.5.2.2 点位设置 I-26
I.5.3 配方信息 I-27
I.5.4 PC 端串接与上传至第三方系统设置 I-28
I.5.4.1 PC 端 Http 串行接口设置 I-28
I.5.4.2 上传至第三方系统设置 I-29
I.5.4.3 PC 设备数据上传说明 I-29
I.5.4.4 PC 设备数据上传 HTTP 接口说明 I-30
I.5.4.5 PC 设备生产配方请求 HTTP 接口说明 I-31

第1章 产品介绍

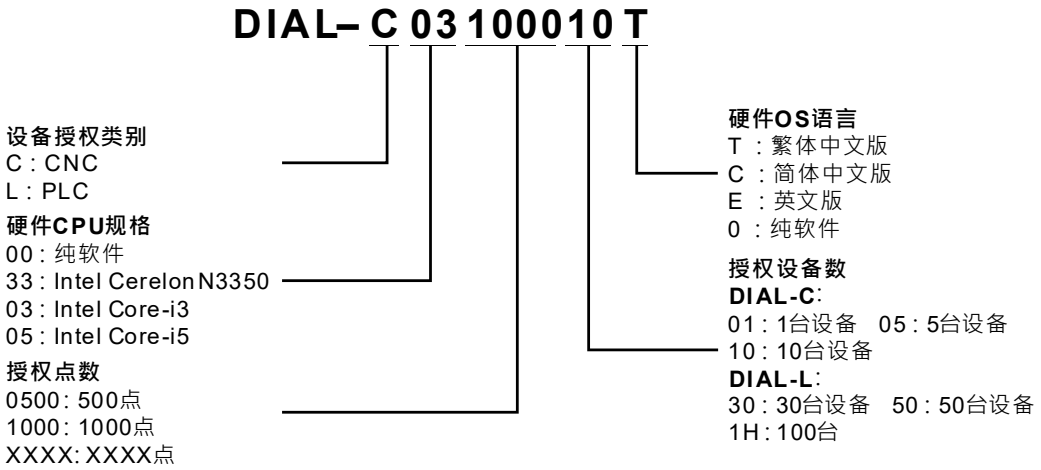
目录

1.1 概述 1-2
1.2 DIALink 机种说明 1-2
1.3 软件版 DIALink 1-4
1.3.1 完全安装 DIALink 1-6
1.3.2 更新 DIALink 1-17
1.3.3 临时软件授权 1-19
1.4 硬件版 DIALink - Intel Core-i 系列主机硬件规格 1-22
1.4.1 外观尺寸图 1-23
1.4.2 I / O 接口示意图 1-24
1.4.3 COM 引脚定义（DB9） 1-25
1.5 硬件版 DIALink - Intel Celeron N3350 系列主机硬件规格 1-26
1.5.1 外观尺寸图 1-27
1.5.2 I / O 接口示意图 1-28
1.5.3 COM 引脚定义（DB9） 1-29
1.5.4 设定 COM 类型 1-30
1.5.5 DIO 引脚定义（DB9） 1-32
1.5.6 设定 DIO 状态 1-32

1.1 概述

DIALink 是一款设备数据采集平台，可有效管理各大厂牌的 CNC 控制器、PLC 等，收集现场设备数据并以统一接口与上层管理平台对接，同时提供可视化信息以反映制程参数与设备工作状态，可搭配台达产品 DIAView 工业图控系统与 OEE 设备管理平台，进行机台加工速度、稼动率以及良率的分析与管理，以实现优化生产排程、预防加工质量异常等成效，从而有效提高设备整体使用率，以边缘运算（Edge Computing）能力将物联网概念实现于工厂自动化。

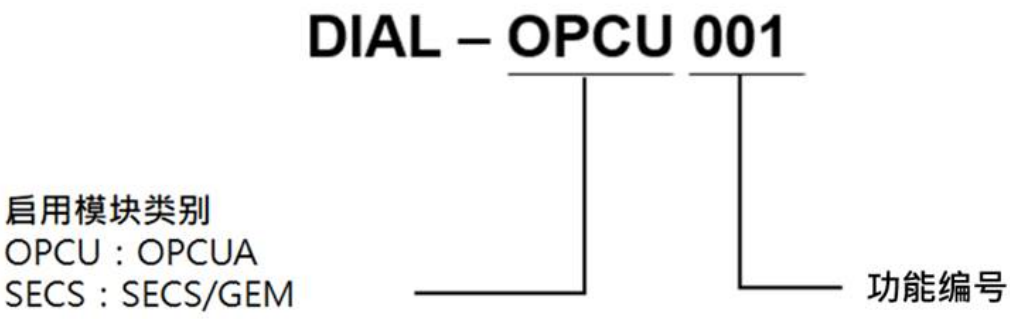
1.2 DIALink 机种说明



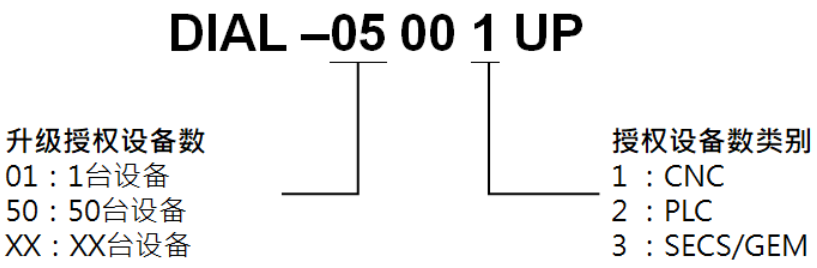
DIALink 产品分为软件版与搭配嵌入式主机的硬件版，软件产品的软件安装与更新流程参考第 1.3 节；硬件版的规格参考第 1.4 节与第 1.5 节，各个产品型号与规格如下：

版本	型号	授权设备数		硬件	
		CNC	PLC	CPU	操作系统
软件版	DIAL-C00050005	5	10	-	-
	DIAL-C00100010	10	20	-	-
	DIAL-L00100030	-	30	-	-
	DIAL-L00200050	-	50	-	-
	DIAL-L0050001H	-	100	-	-
硬件版	DIAL-C03050005C	5	10	Intel Core-i3	Win10 Pro 简体中文版
	DIAL-C03050005T				Win10 Pro 繁体中文版
	DIAL-C03050005E				Win10 Pro 英文版
	DIAL-C03100010C	10	20		Win10 Pro 简体中文版
	DIAL-C03100010T			Intel Core-i3	Win10 Pro 繁体中文版
	DIAL-C03100010E				Win10 Pro 英文版

版本	型号	授权设备数		硬件	
		CNC	PLC	CPU	操作系统
硬件版	DIAL-L03100030C	-	30	Intel Core-i3	Win10 Pro 简体中文版
	DIAL-L03100030T				Win10 Pro 繁体中文版
	DIAL-L03100030E				Win10 Pro 英文版
	DIAL-L03200050C	-	50		Win10 Pro 简体中文版
	DIAL-L03200050T				Win10 Pro 繁体中文版
	DIAL-L03200050E				Win10 Pro 英文版
	DIAL-L0550001HC	-	100	Intel Core-i5	Win10 Pro 简体中文版
	DIAL-L0550001HT				Win10 Pro 繁体中文版
	DIAL-L0550001HE				Win10 Pro 英文版
	DIAL-C33010001E	1	2		Win10 IoT 英文版



版本	型号	启用模块类型	
		OPCUA	SECS/GEM
软件版	DIAL-OPCU001	●	-
	DIAL-SECS001	-	●



版本	型号	升级授权设备数		
		CNC	PLC	SECS/GEM
软件版	DIAL-05001UP	5	10	-
	DIAL-10001UP	10	20	-
	DIAL-30002UP	-	30	-
	DIAL-50002UP	-	50	-
	DIAL-01003UP	-	-	1
	DIAL-02003UP	-	-	2
	DIAL-03003UP	-	-	3
	DIAL-04003UP	-	-	4

1.3 软件版 DIALink

适用机种	● 授权设备类型：CNC DIAL-C00050005、DIAL-C00100010 ● 授权设备类型：PLC DIAL-L00100030、DIAL-L00200050、DIAL-L0050001H ● DIAL-P 授权升级为 DIAL-L（设备类型：PLC） DIAL-U00100030、DIAL-U00200050、DIAL-U0050001H ※DIAL-P 为 DIALink Windows AP 版，DIAL-L 为 DIALink Web 版。

请用户根据软件盒内安装手册的台达指定网址下载 DIALink 安装档。

● 注意事项

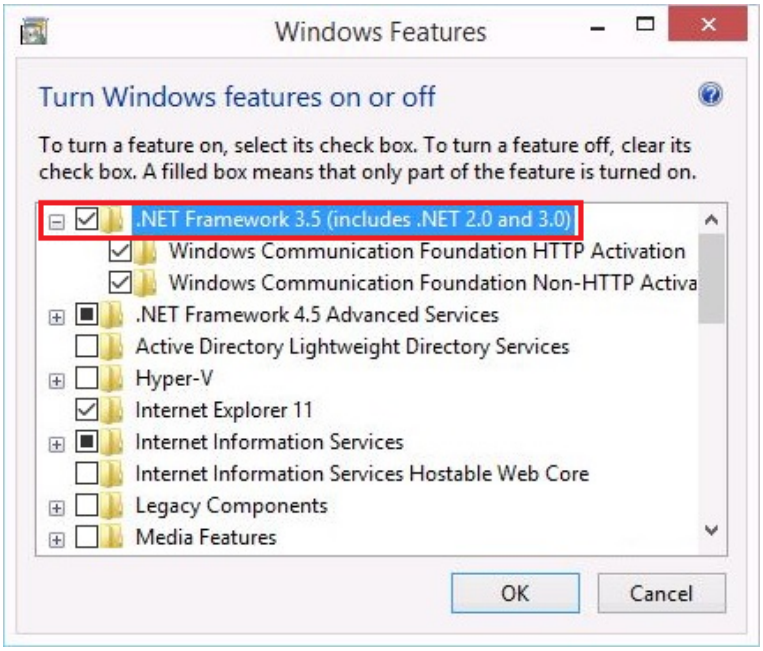
1. 安装前请确认计算机操作系统符合需求。
2. 应依正确用途使用软件，不得破解或另作它用。
3. 建议作业环境需求

操作系统	Windows 7 x86 /x64 SP1　Professional 或更高 Windows 10 x86 /x64　Professional 或更高
最小内存需求	4GB
最小硬盘可用空间需求	256GB

4. 安装 DIALink 的主机内，应避免安装第三方软件，以免影响主机效能与软件兼容性。

5. 安装 DIALink 前，主机必须至【控制面板】→【程序集】→【开启或关闭 Windows 功能】内开

启 Windows 内建的【.Net Framework 3.5】功能（主机需联网）。



6. 仅下载软件试用机制，其限制为 1 台设备、10 个点位读取及 30 天试用期。建议消费者购买正式版，以便取得较多的设备、点位及较长的使用期限。

1

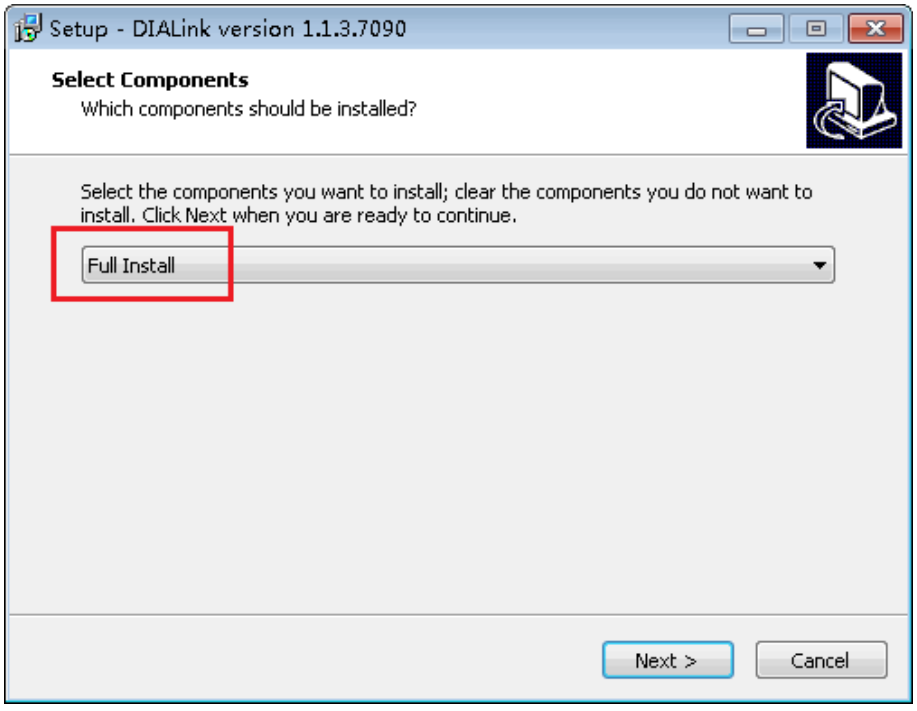
1.3.1 完全安装 DIALink

若客户使用环境对资安有严格的要求，限定不论是安装或者是移除，甚至是通过 DIALink.APPSettingsConfigTool 变更设定参数，都需要具备管理员权限，客户安装或移除以及变更设定前请务必先向 IT 部门取得管理员权限以实时进行。

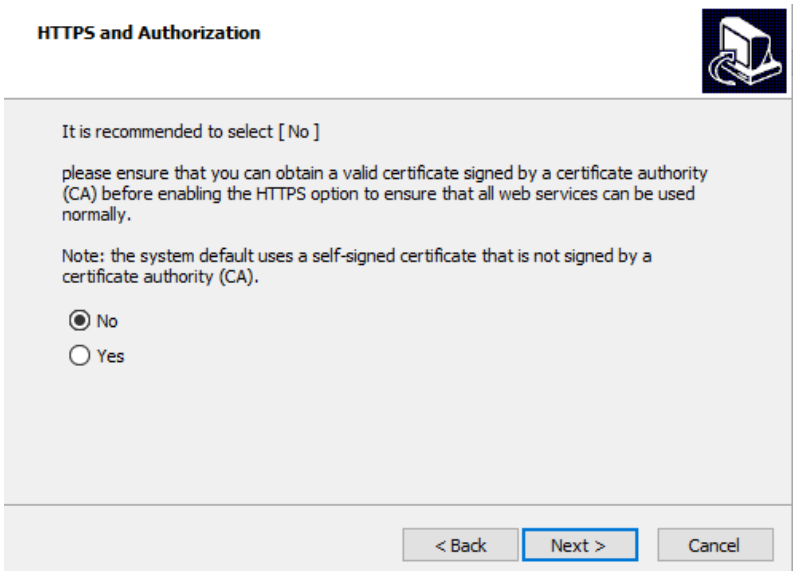
Step 1. 鼠标左键双击下载的文件，执行 DIALink 一键安装档。



Step 2. 下拉式选单选择【Full Install】。



Step 3. HTTPS and Authorization 设定页面



- No：(不启用)

  默认 DIALink Web 服务(包含登入网页、Web API)均采用 HTTP 通讯，所有提供第三方程式调用的 Web API 除方法 GET （取得资源）外，需先取得 Token 授权。

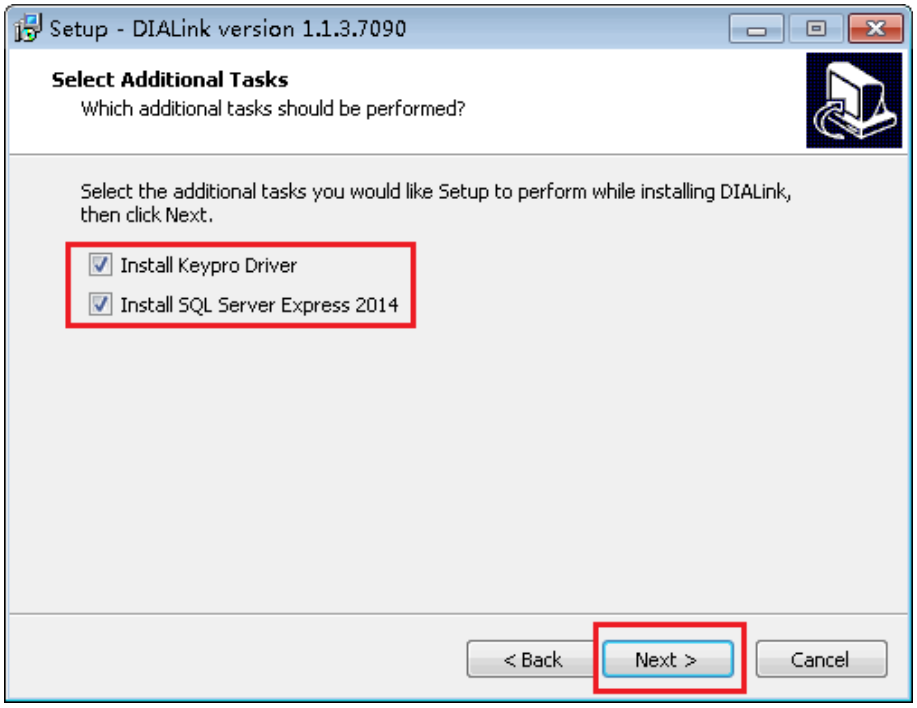
- Yes：(启用)

  DIALink Web 服务(包含登入网页、Web API)会采用 HTTPS通讯，请确保能取得证书颁发机构（CA）签署的合法凭证才启用 HTTPS 选项，以确保所有 Web 服务能正常使用 (手动更新请参考附录 H)。所有提供第三方程式调用的 Web API 需先取得 Token 授权。

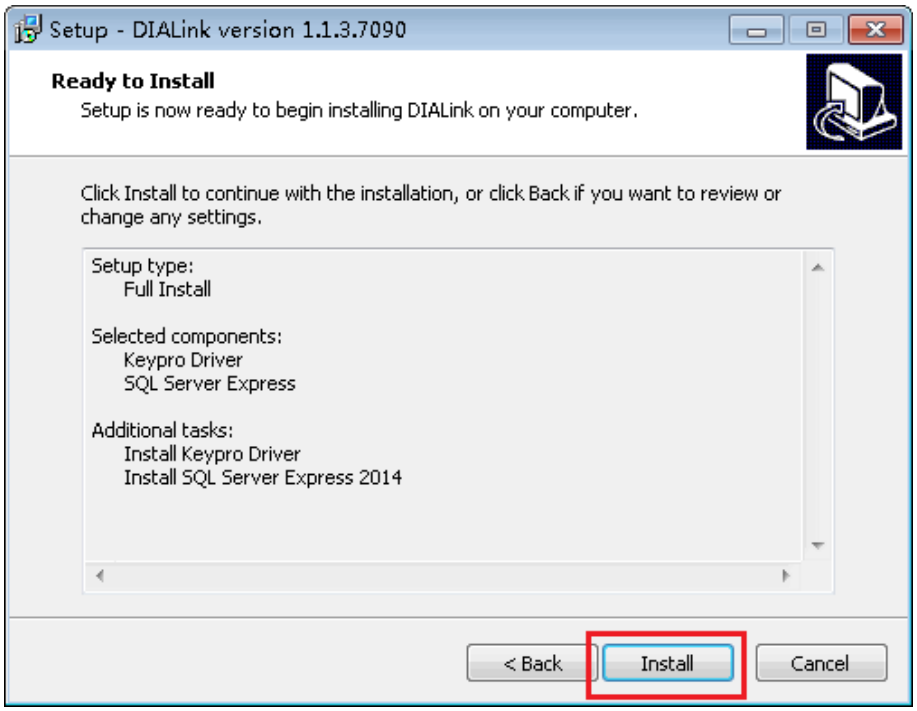
  注释:

  自签凭证和 CA 签名证书一样可以用来加密数据，但是由于该凭证不属于公钥基础设施签发凭证，其他没信任该凭证的 HTTPS 存取者的网页浏览器会显示一个警告，说这个凭证不被他们的计算机或浏览器信任，必须在进阶设定后才能浏览使用该网页。

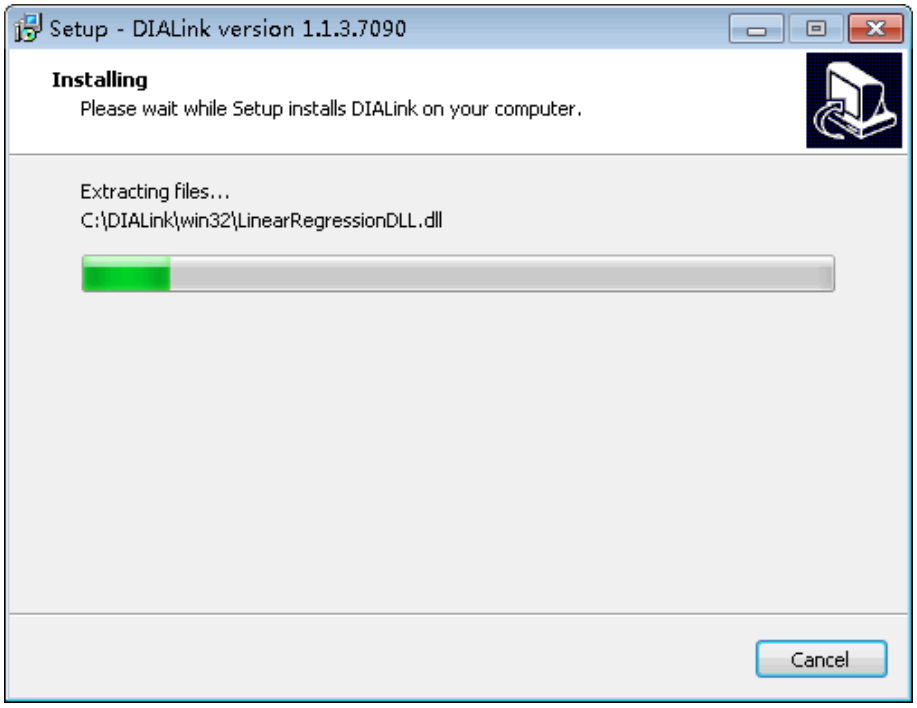
Step 4. 选择安装 DIALink 的附加程序，第一选项【Install Keypro Driver】为加密锁驱动，第一次安装必须点选；DIAL 会将点位数据保存在 SQL Server，若用户未安装 SQL，必须单击第二选项【Install SQL Server Express 2014】。以上两选项皆为默认安装，单击【Next】继续下一步骤。



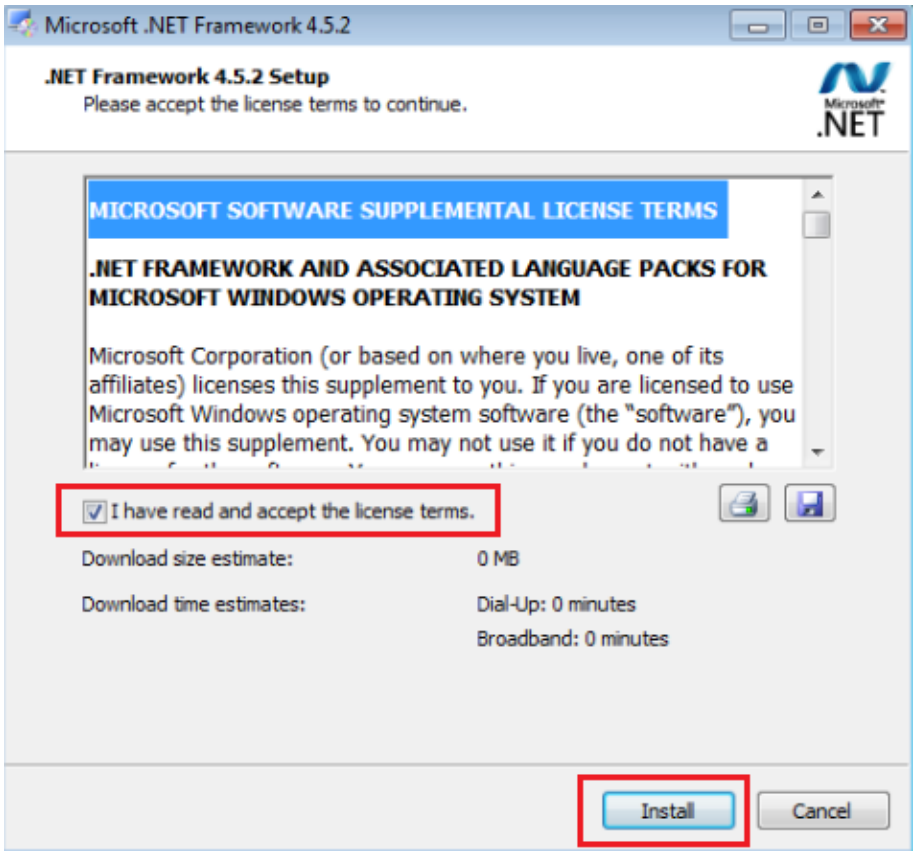
Step 5. 再次确认安装内容，单击【Install】继续下一步骤。



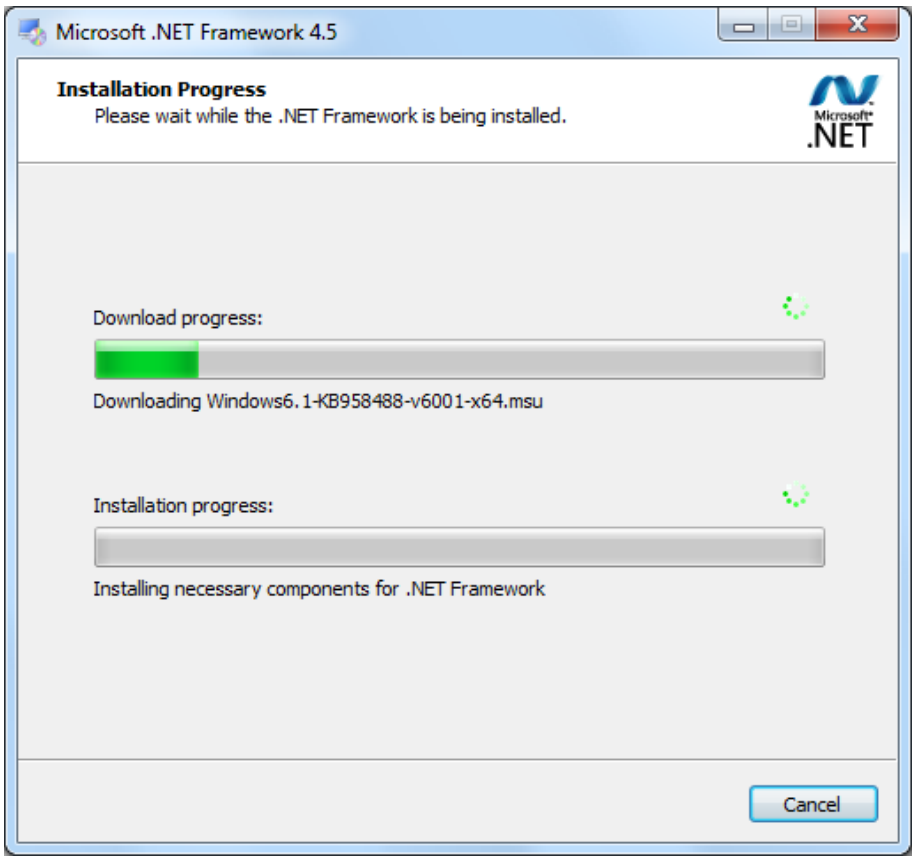
Step 6. 等待安装 DIALink 软件，安装完毕会自动开启下一步骤。



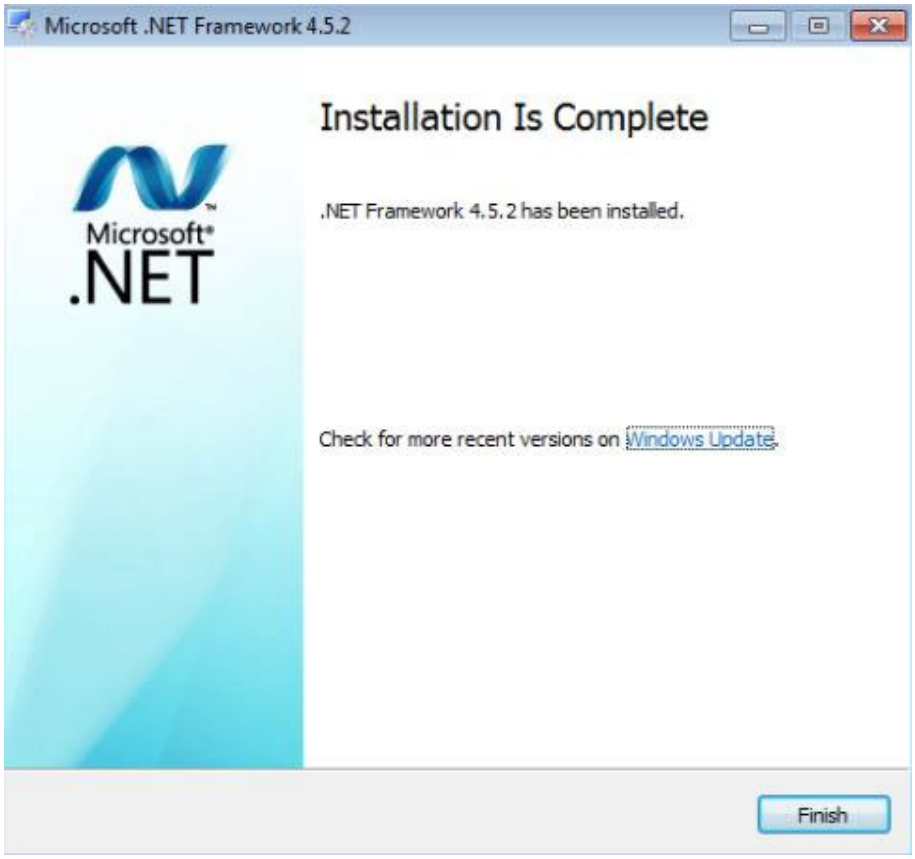
Step 7. 若主机尚未安装.Net Framework 4.5 套件，会自动启动安装，请详细阅读授权条款后，勾选已阅读授权条款选项，接着单击【Install】继续下一步骤。



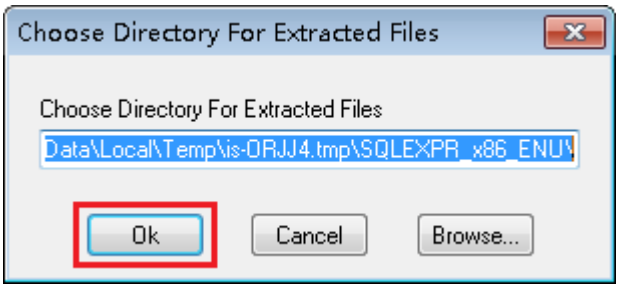
Step 8. 等待安装.Net Framework 4.5 套件。



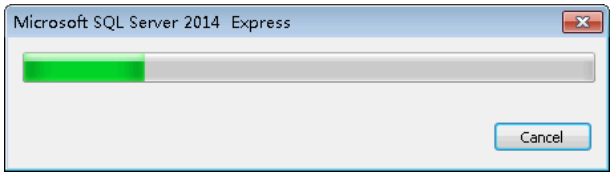
Step 9. 完成安装.Net Framework 4.5 套件，单击【Finish】继续下一步骤。



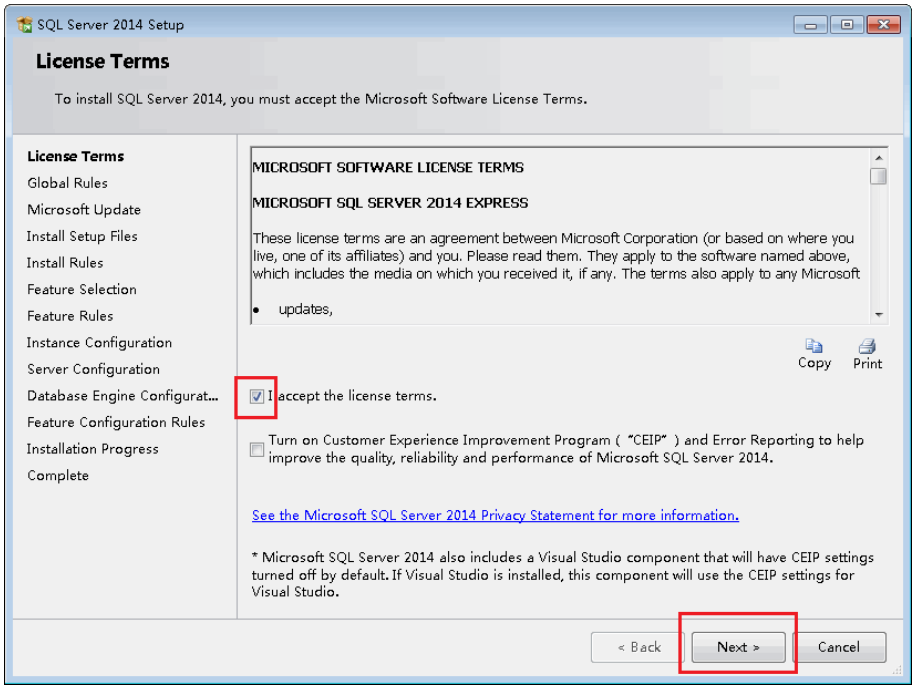
Step 10. 若 Step3 有选择安装 SQL Server Express，会自动启动安装 SQL，选择执行文件的暂存盘路径，建议维持默认，直接单击【OK】继续下一步骤。



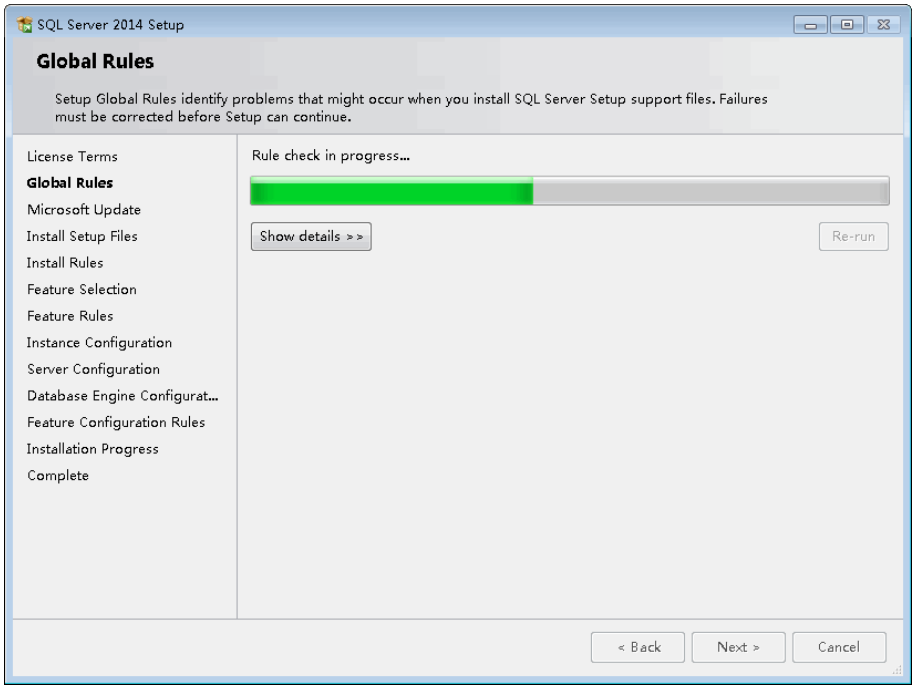
Step 11. 等待 SQL 执行档解压缩。



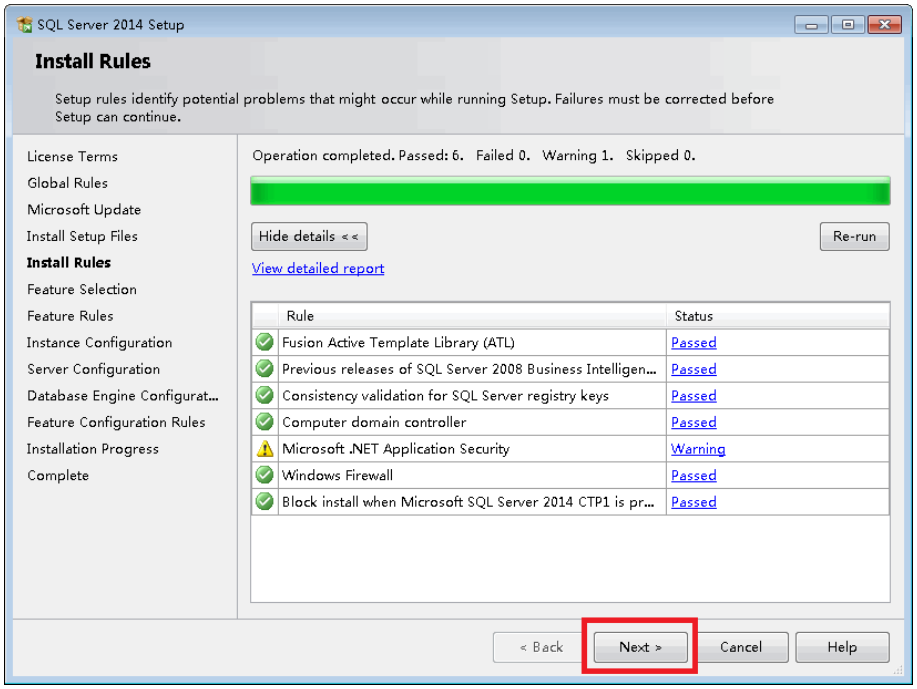
Step 12. 请详细阅读授权条款后，勾选同意授权条款选项，接着单击【Next】继续下一步骤。



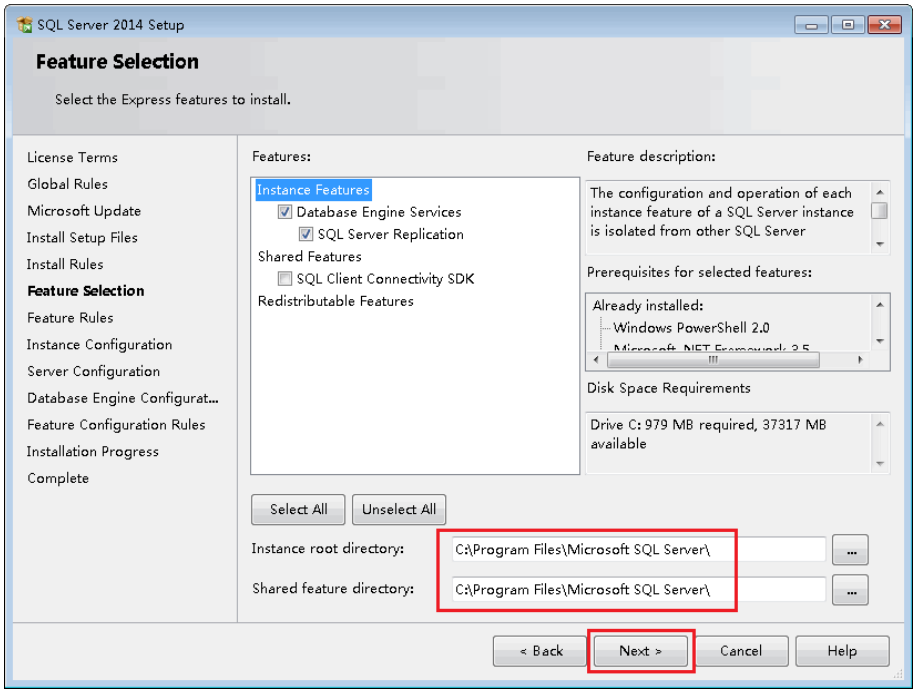
Step 13. 等待 SQL 执行程序。



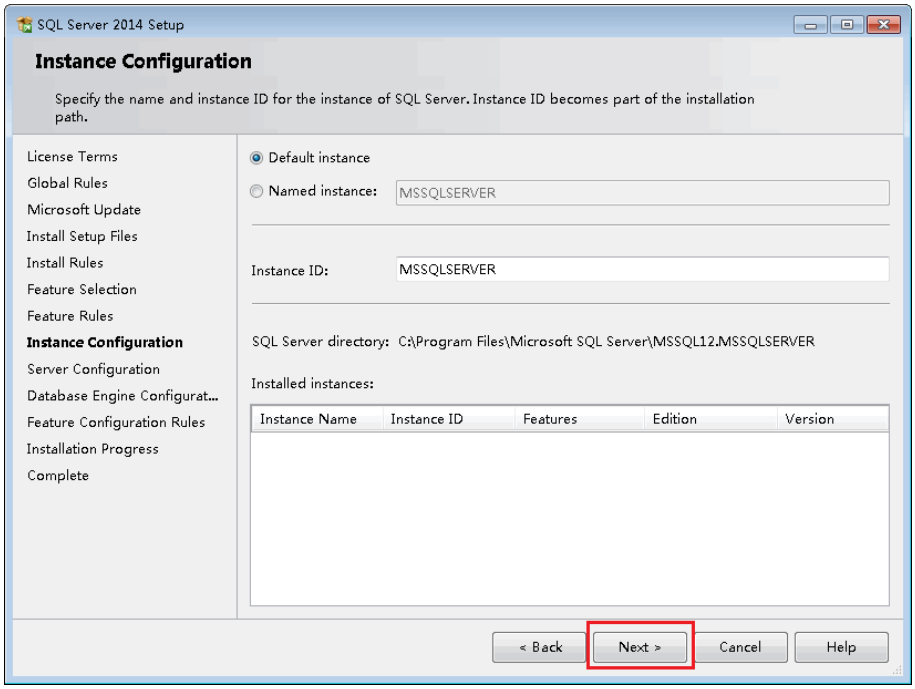
Step 14. 单击【Next】继续下一步骤。



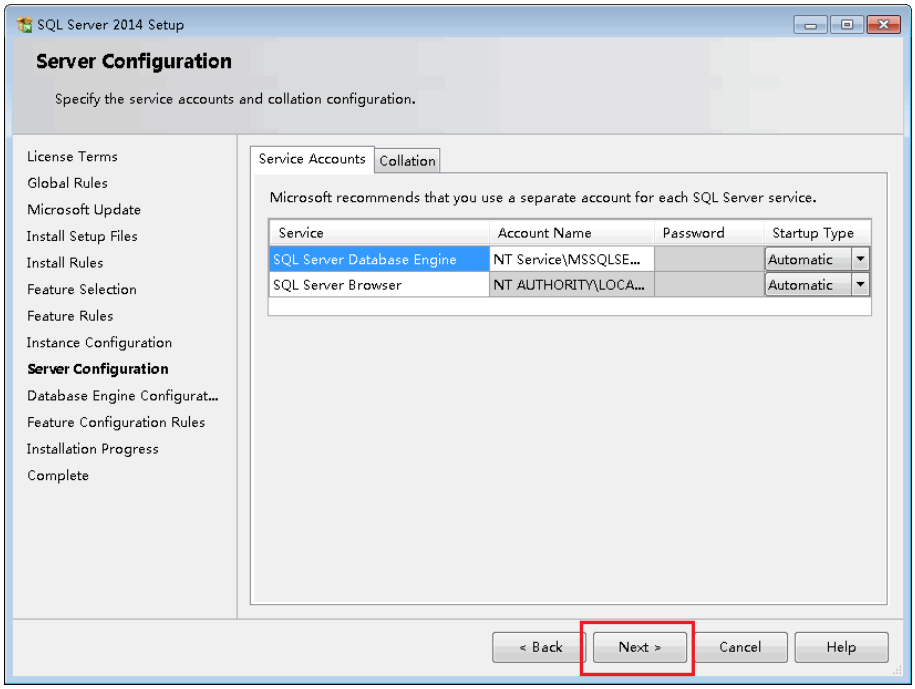
Step 15. 选择 SQL 安装路径，必须确保磁盘区有足够空间，接着单击【Next】继续下一步骤。



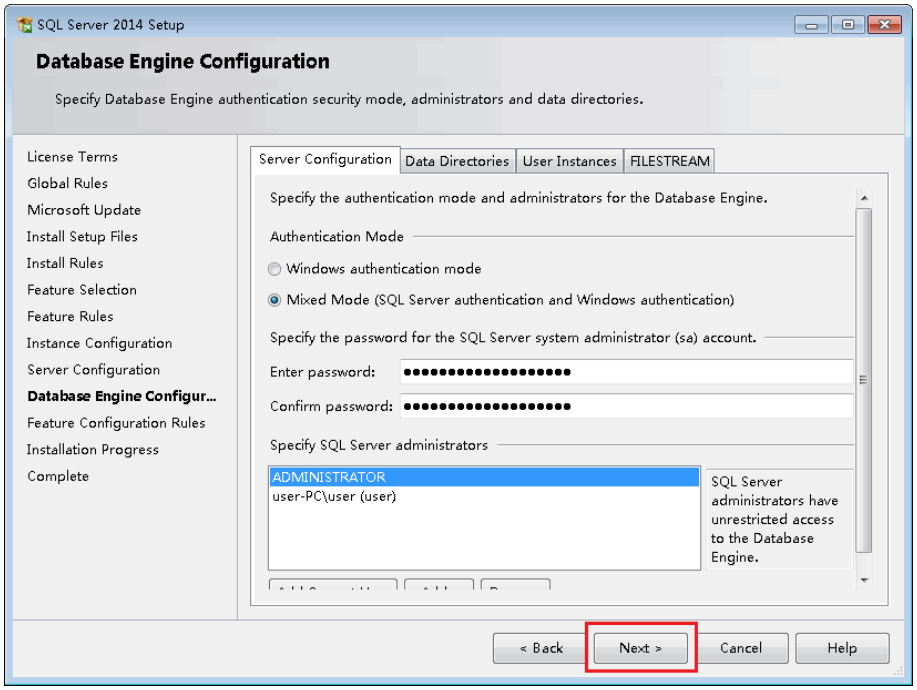
Step 16. 此为默认页面，单击【Next】继续下一步骤。



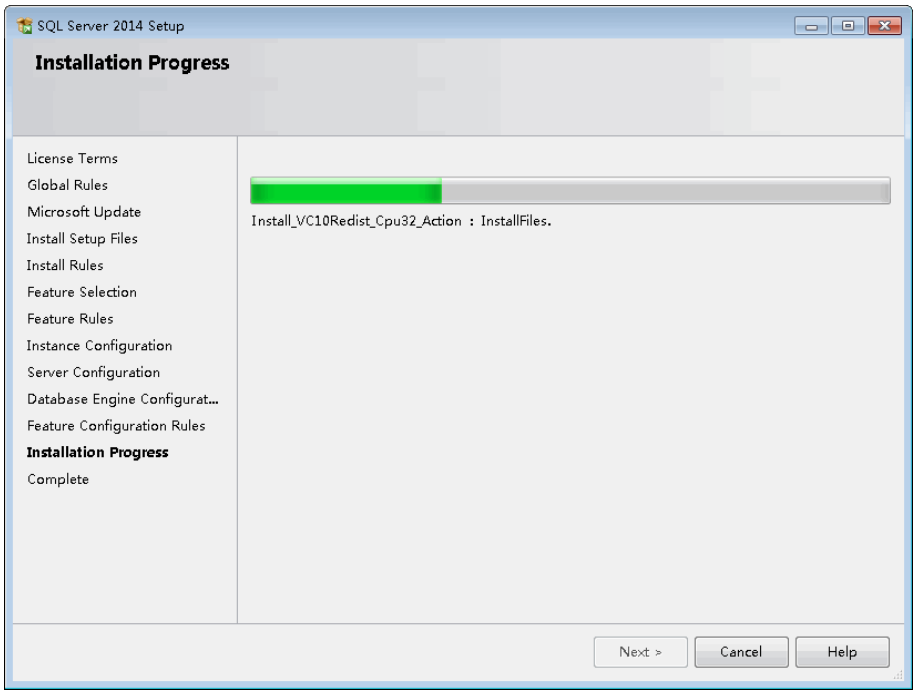
Step 17. 此为默认页面，单击【Next】继续下一步骤。



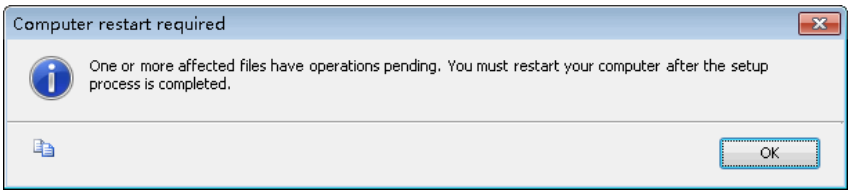
Step 18. 此为默认页面，单击【Next】继续下一步骤。



Step 19. 等待套件安装 SQL Server Express 软件。

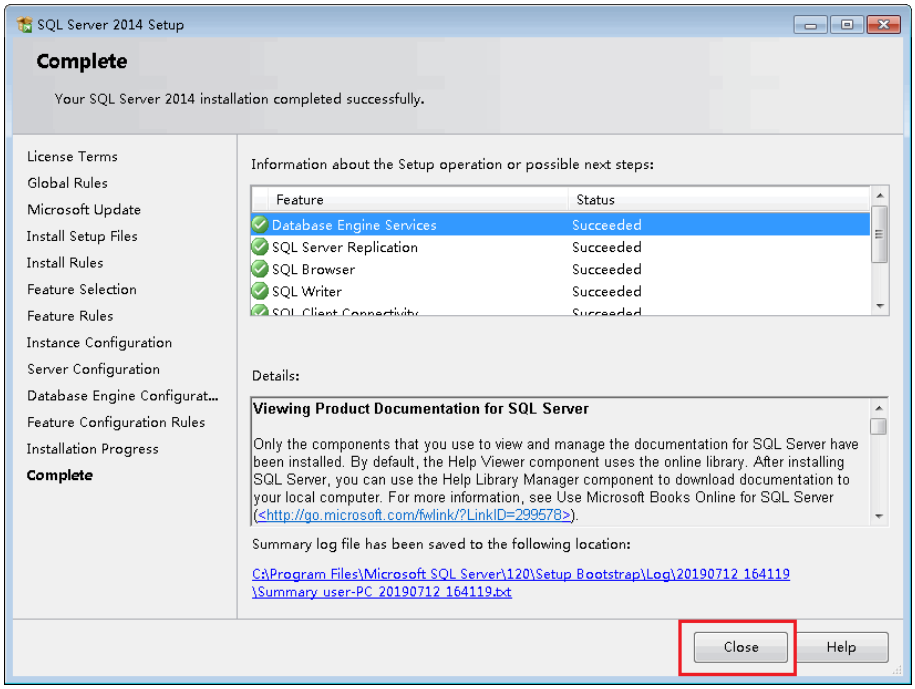


Step 20. SQL 安装结束前可能会出现需要重新开机的提示。

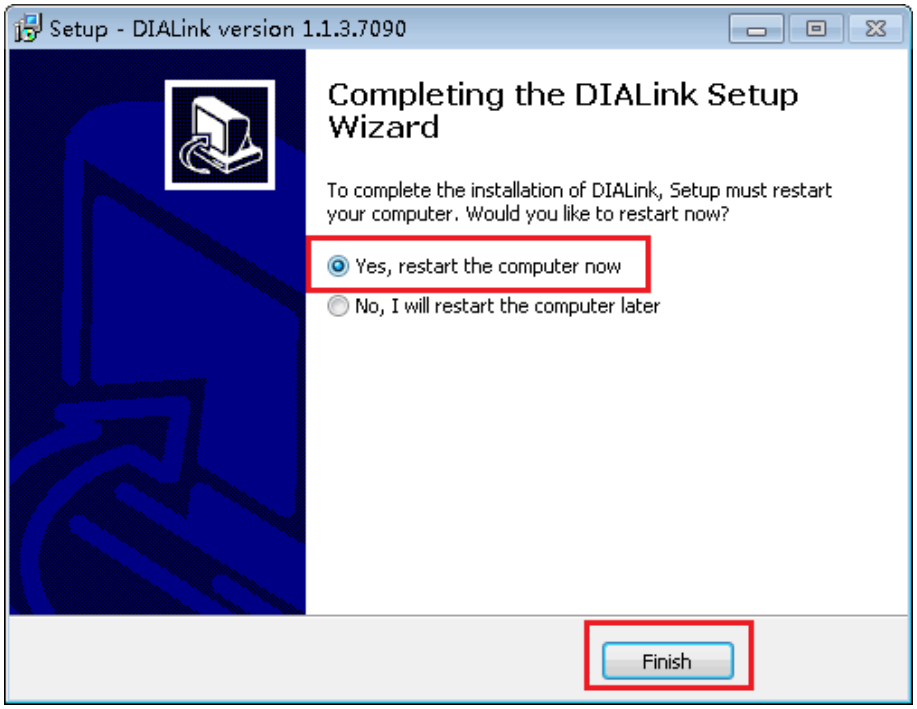


1

Step 21. 完成安装 SQL Server Express，单击【Close】关闭画面。



Step 22. 单击【Finish】，重新启动主机，即可完成安装，桌面会出现名为 DIALink 的网页快捷方式。

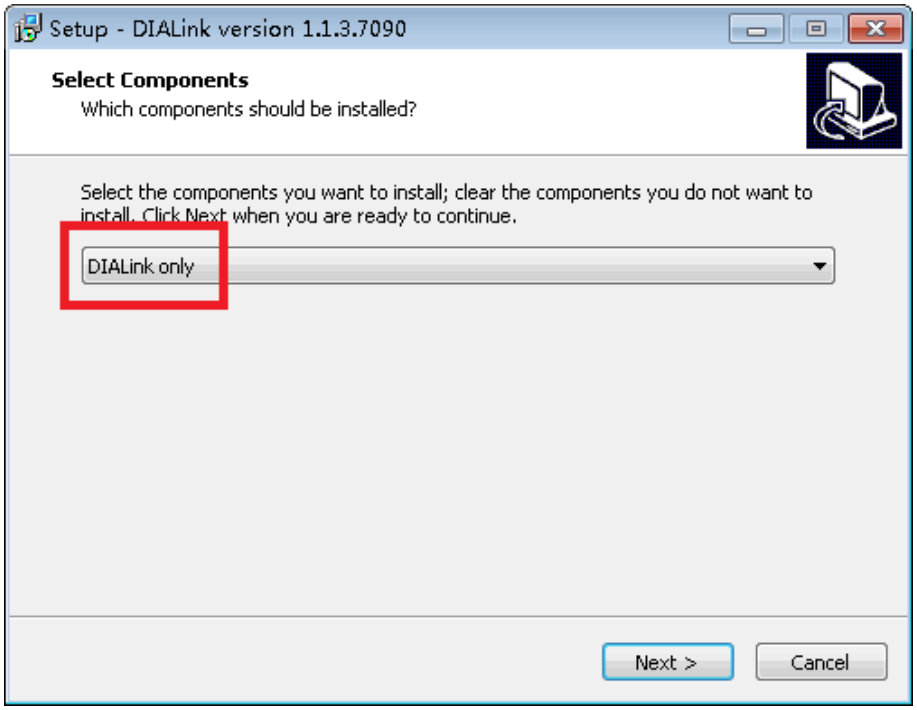


1.3.2 更新 DIALink

Step 1. 鼠标左键双击下载的文件，执行 DIALink 一键安装档。

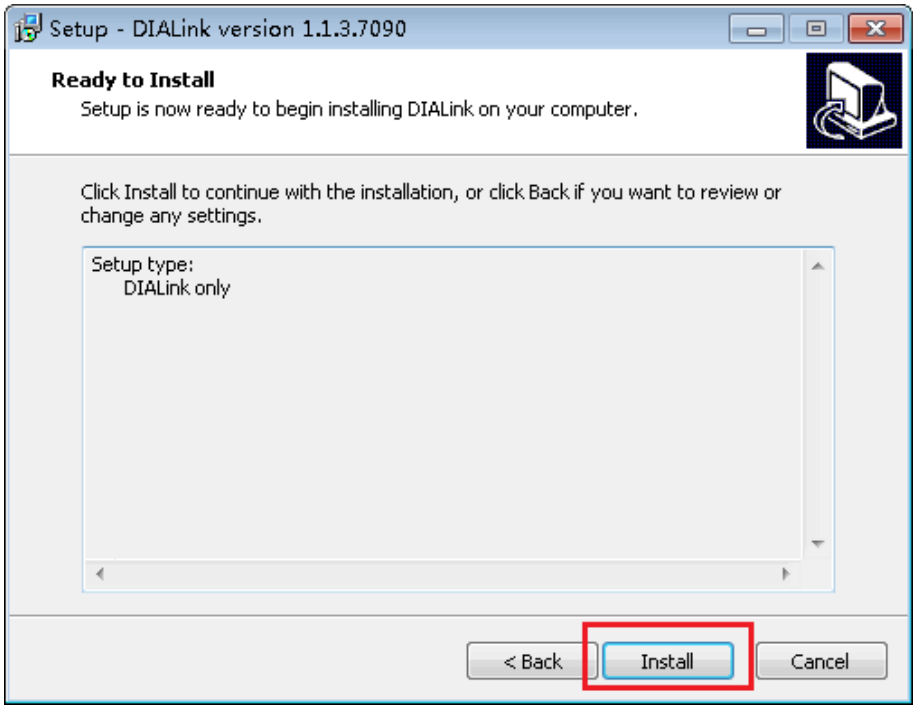


Step 2. 下拉式选单选择【DIALink Only】。

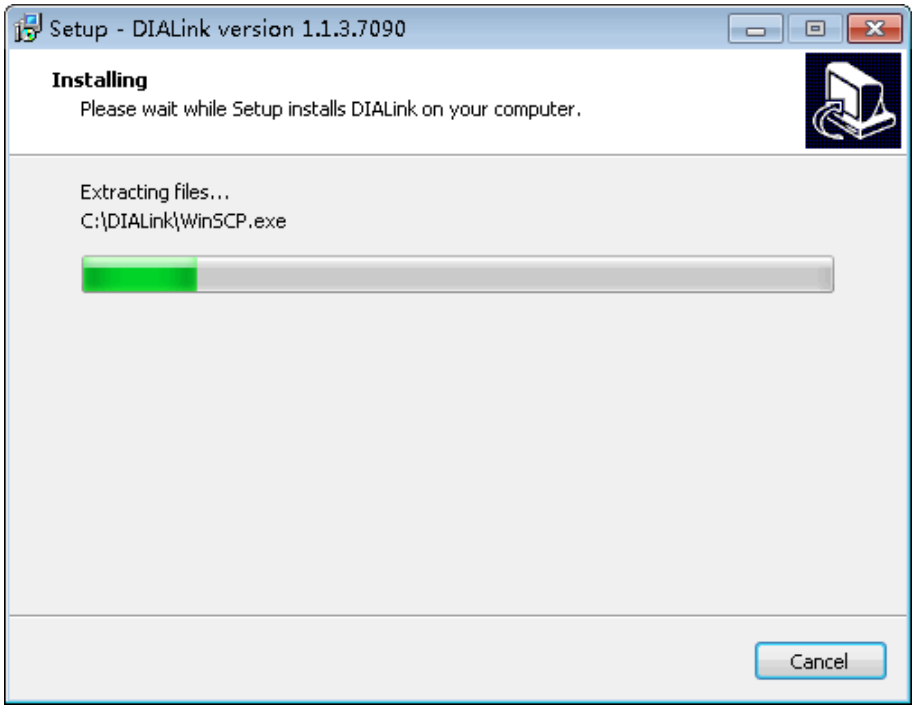


1

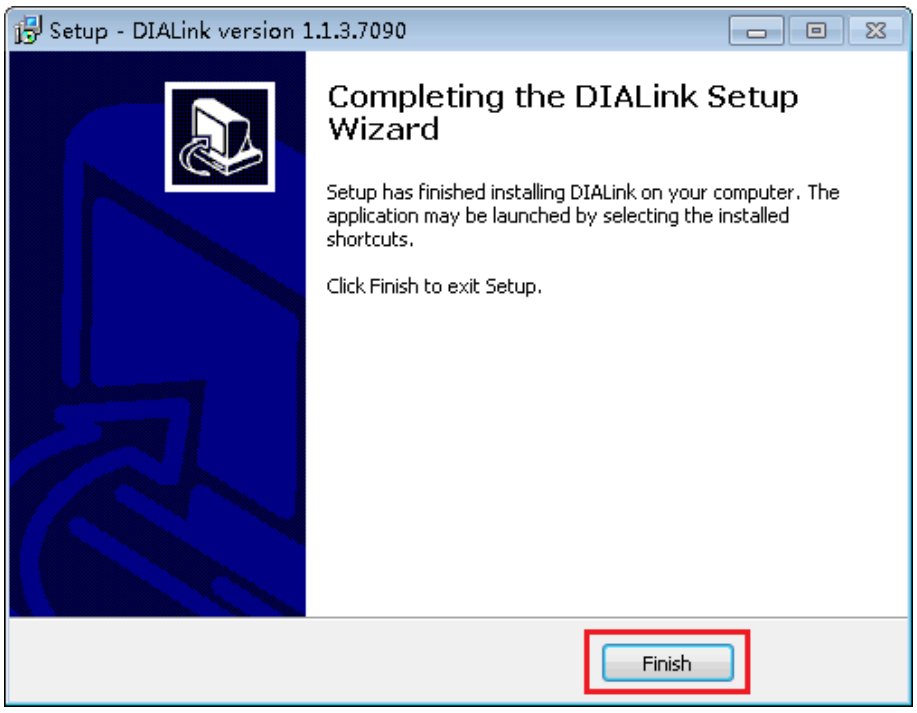
Step 3. 默认页面，单击【Install】继续下一步骤。



Step 4. 等待安装 DIALink 软件。



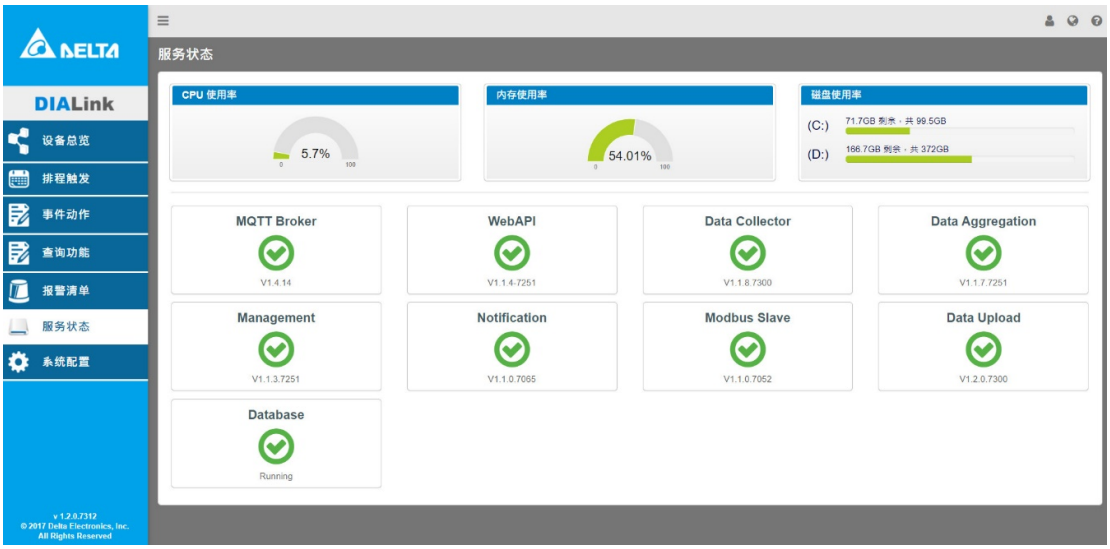
Step 5. 单击【Finish】完成 DIALink 软件安装。



1.3.3 临时软件授权

1. DIALink 软件授权流程：
   - A 下载 DIALink 安装包 。
   - B 对 DIALink 安装包右键以**系统管理员身分执行**。
   - C 依照计算机是否安装 DIALink 选择完全安装或 DIALink Only。

2. **安装成功与否确认：**
   - A 安装完成后开启浏览器打开 127.0.0.1:5000，出现 DIALink 的登入画面(默认帐号密码 root/admin)。



B　所有服务状态为绿色勾表示运作正常。

C　安装完成可使用试用机制（1 台设备联机、10 个点位读取及 30 天试用的限制），**需取得软件临时授权或是购买正版加密锁**，才可以新增超过一台设备。



3. **软件临时授权：**

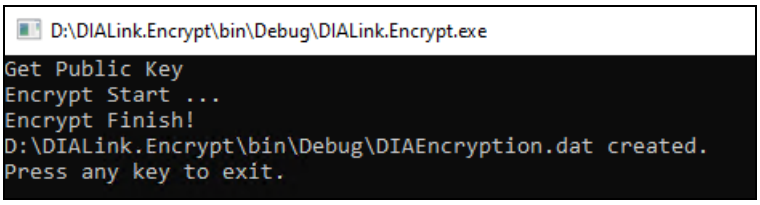
A　请以 E-Mail 或电话向经销商或 FAE 工程人员申请软件试用，申请书模板如下。

1. Applicant: _____
2. Unit: _____
3. Application type: CNC or PLC
4. Equipment Quantity:
5. CNC: 5 or 10
   PLC: 30 or 50 or 100
6. SECS: 1 ~ 5
7. OPCUA Server □ need □ unneeded
8. Purpose:_____



B 确认申请后会收到回信 信件附件 DIALink.Encrypt 为主机信息注册工具 ,请在安装 DIALink 的计算机将此文件解压缩后，右键-使用系统管理员身分执行 DIALink.Encrypt.exe。

C 若执行后正确则会可看到此 Finish 窗口。



请将 DIALink.Encrypt.dat 文件以信件附件方式回传经销商或 FAE 工程人员。很快就会收到软件适用授权文件：DIALinkLicense.dat 的回信，将此附件文件放置于 C:\DIALink 即可。

1

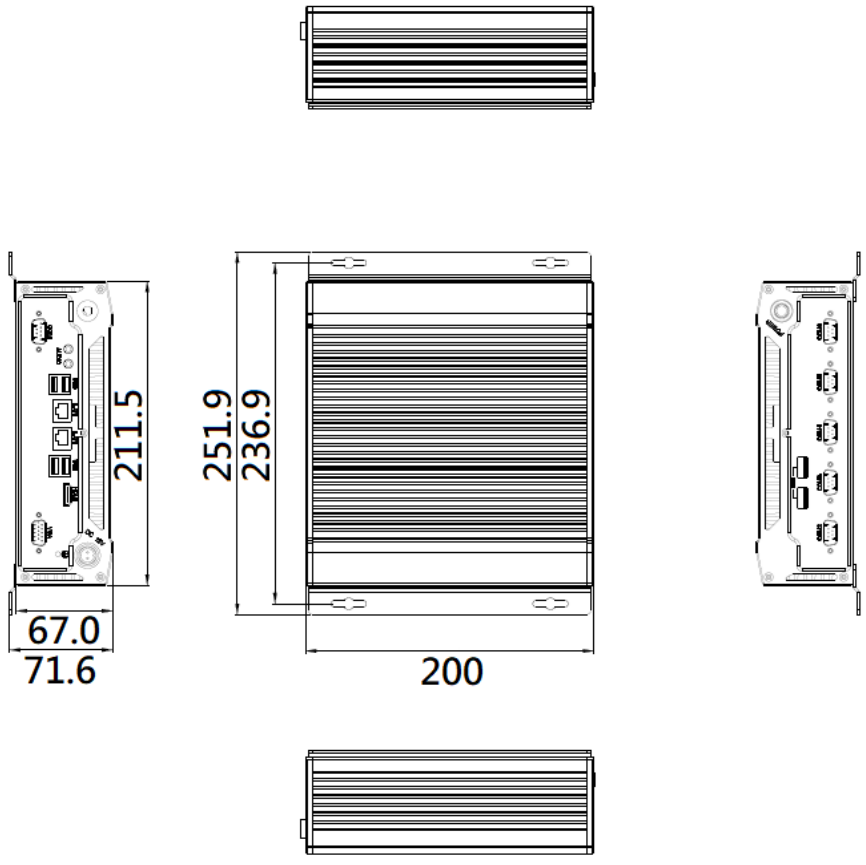
1.4 硬件版 DIALink - Intel Core-i 系列主机硬件规格

适用机种	● **Core-i3 主机（DIAVH-IPC00310xA）** DIAL-C03050005T、DIAL-C03100010T、DIAL-L03100030T、DIAL-L03200050T ● **Core-i5 主机（DIAVH-IPC00510xA）** DIAL-L0550001HT

项目	规格
中央处理器	Intel ® 第八代 Core i3-8145U / i5-8265U 处理器
内存	4GB DDR4-2133/2400，最大支持 16GB
硬盘	256GB mSata SSD
网络功能	Intel® I211 Gigabit Ethernet Controller x2
音频功能	一组标准音频输出
扩展接口	支持 mini PCIE 接口，可扩展 Wi-Fi 等无线通信模块
外接 I/O	● 串行通讯口：COM 1 ~ 2 & 5 ~ 6（RS-232）、COM 3 ~ 4（RS-485） ● USB 2.0 x2、USB 3.0（蓝色）x6 ● VGA 输出 x1 ● HDMI 输出 x1
外型尺寸	不含支架：211.5mm（长）× 200mm（宽）× 67mm（高）; 含支架：251.9mm（长）× 200mm（宽）× 71.6mm（高）
结构	● 机壳：高强度优质铝合金，喷砂阳极氧化颜色 ● 机壳颜色：黑色 ● 导热：一体式全铝机身主动式导热设计 ● 材质：高强度铝合金散热板
重量	净重：2.8Kg；毛重：3.5Kg
温度	● 工作温度：-20°C ~ 60°C （表面空气流动） ● 储存温度：-40°C ~ 70°C
湿度	95% @ 40°C（非凝结状态）
电磁兼容性	● 无线电干扰限值符合 GB9254-1998 标准 A 级 ● 抗扰度符合 GB/T 17618 标准规定的限值
可靠性	● 平均无故障工作时间：MTBF ≥ 5000h ● 平均维修时间：MTTR ≤ 0.5h
认证	CE、FCC、BSMI

项目	规格
机械环境适应性	● 抗振动：5-19Hz / 1.0mm 振幅；19-200Hz / 1.0g 加速度 ● 抗冲击：10g 加速度，11ms 周期
电源特性	● 标配变压器： 输入电压 / 频率：100~240VAC / 50~60Hz 输出电压 / 电流：12VDC / 5A ● 整机功耗：7W（待机状态） ● 整机功耗：16W（运行 Max Power 100%）

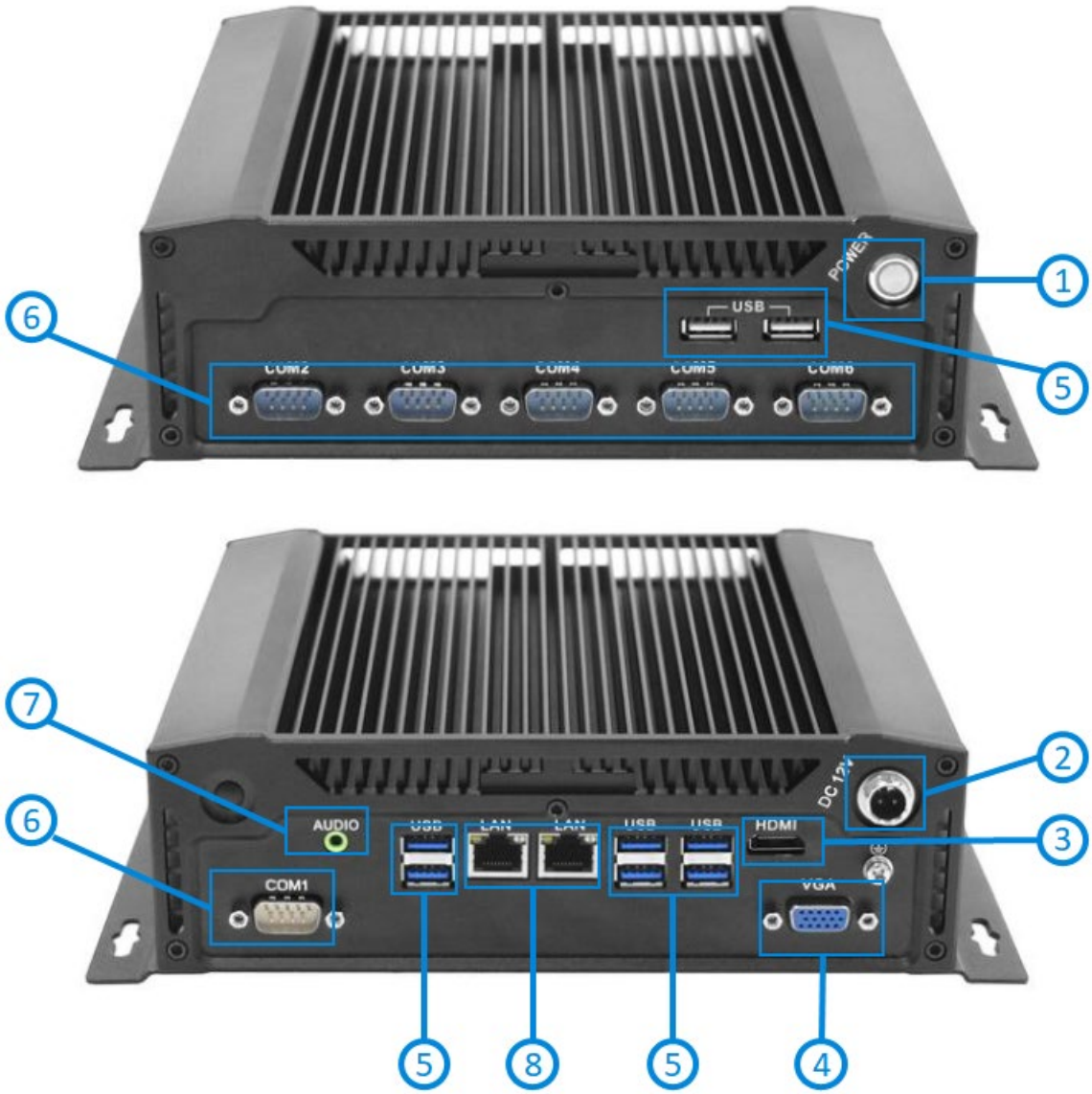
1.4.1 外观尺寸图



单位：mm

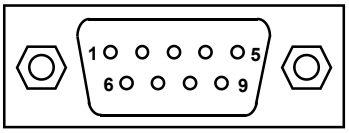
1

1.4.2 I / O 接口示意图



No.	Item	No.	Item
1	Power Switch	5	USB2.0 / USB3.0
2	Power Connector	6	RS-232 / RS-485
3	HDMI	7	Line-Out
4	VGA	8	Ethernet

1.4.3 COM 引脚定义（DB9）



脚位	通讯接口	
	RS-232	RS-485
1	DCD	DATA+
2	RXD	DATA-
3	TXD	-
4	DTR	-
5	GND	GND
6	DSR	VCC +5V （备用输入）
7	RTS	-
8	CTS	-
9	RI	-

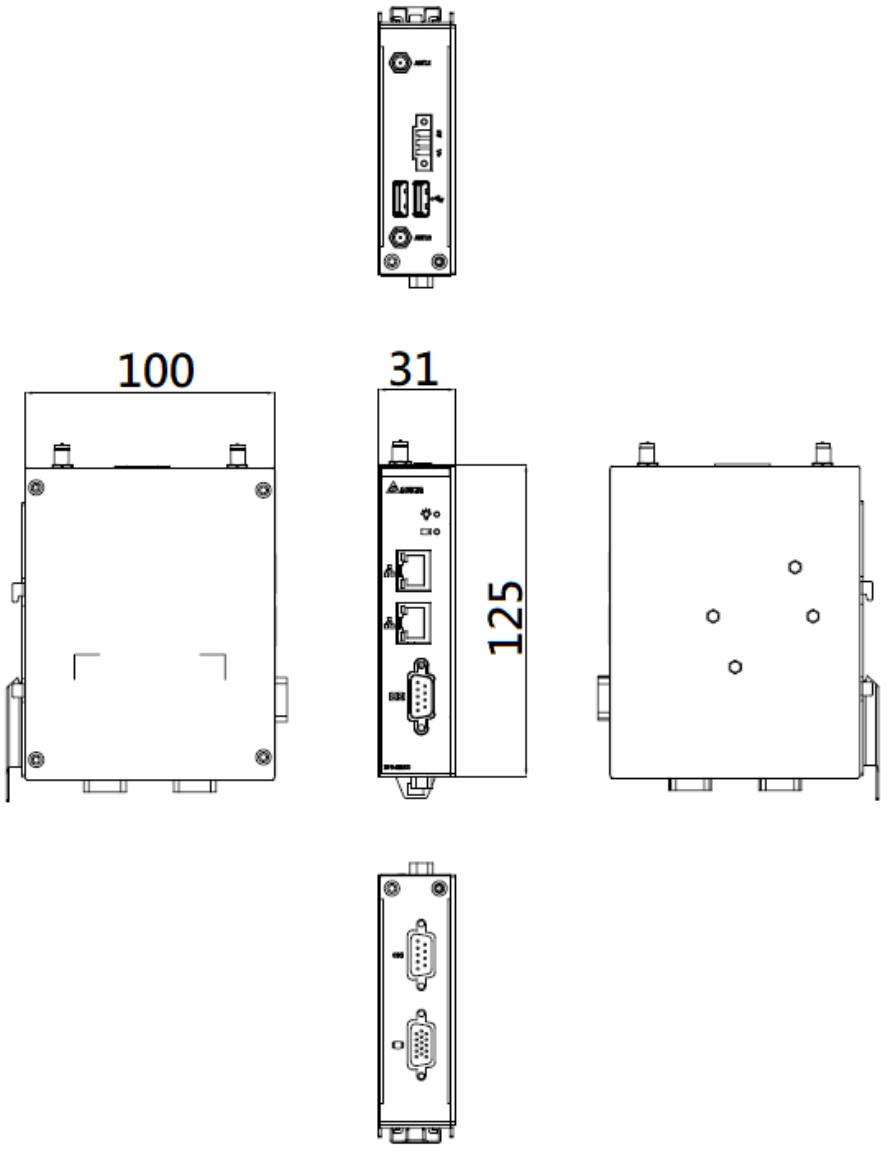
1

1.5 硬件版 DIALink - Intel Celeron N3350 系列主机硬件规格

适用机种	● **嵌入式系统（IPC-E200-N31202E00）** DIAL-C33010001E、DIAL-S33000001E

项目	规格
中央处理器	Intel® Celeron N3350
内存	2GB DDR3L 1600MHz，最大支持 8GB
硬盘	64GB mSata SSD
网络功能	Intel® i211AT Gigabit Ethernet Controller x2
扩展接口	Full-size Mini PCIe，可扩展无线模块
外接 I/O	● RS-232 / RS-422 / RS-485 x1，默认皆为 RS-232，可在 BIOS 内设定，设定方式参考第 1.5.4 节 ● USB 2.0 x2 ● VGA 输出 x1 ● DI/O x1（8-bit，DB9 界面），pin 1 ~ 4 默认为 DO，pin 5 ~ 8 默认为 DI，可在 BIOS 内设定，设定方式参考第 1.5.6 节
外型尺寸	100mm（长）× 31mm（宽）× 125mm（高）
重量	净重：0.3Kg；毛重：0.46Kg
温度	● 工作温度：-20°C ~ 70°C（表面空气流动） ● 储存温度：-45°C ~ 85°C
湿度	10% ~ 95%，非凝结状态
可靠性	● 平均无故障工作时间：MTBF ≥ 69,000h ● 平均维修时间：MTTR ≤ 0.5h
认证	CE、FCC
机械环境适应性	● 抗振动：2.0G，5 ~ 500Hz
电源特性	● 电压输入范围：12~24VDC，17W

1.5.1 外观尺寸图



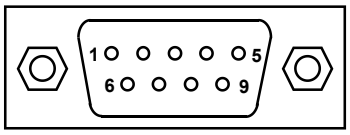
单位：mm

1.5.2 I / O 接口示意图



No.	Item	No.	Item
1	Power Connector	4	RS-232 / RS-422 / RS-485
2	VGA	5	Ethernet
3	USB 2.0	6	DIO

1.5.3 COM 引脚定义（DB9）

Serial Port	脚位	通讯接口		
		RS-232	RS-485	RS-422
	1	DCD	Data-	TX-
	2	RXD	Data+	TX+
	3	TXD	-	RX+
	4	DTR	-	RX-
	5	GND	GND	GND
	6	DSR	-	-
	7	RTS	-	-
	8	CTS	-	-
	9	RI	-	-

1

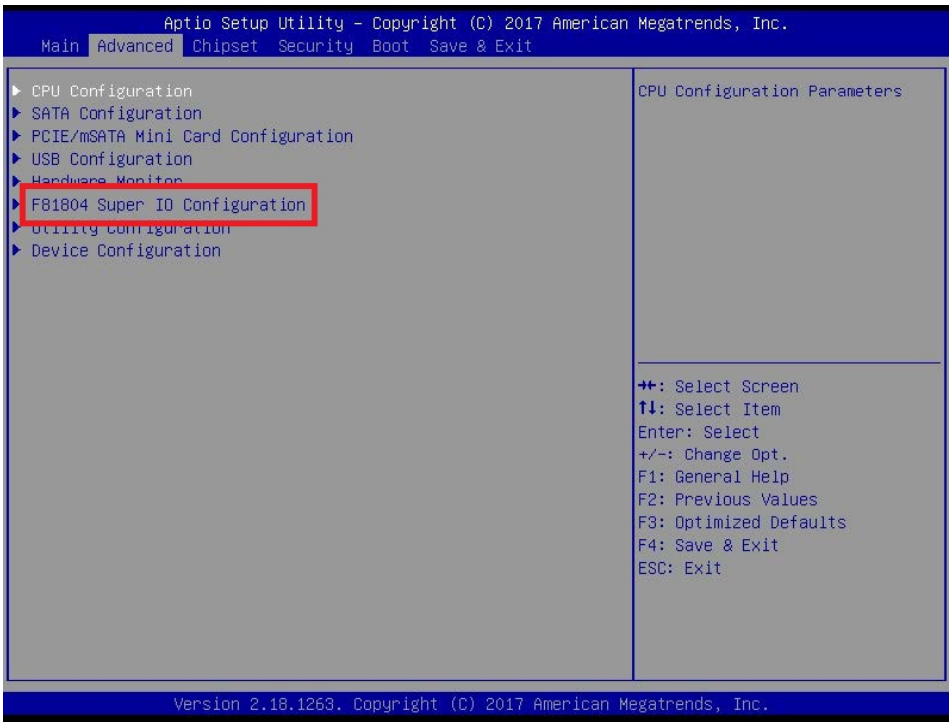
1.5.4 设定 COM 类型

COM 出厂值为 RS-232，若要变更 COM 类型，可参考以下操作步骤：

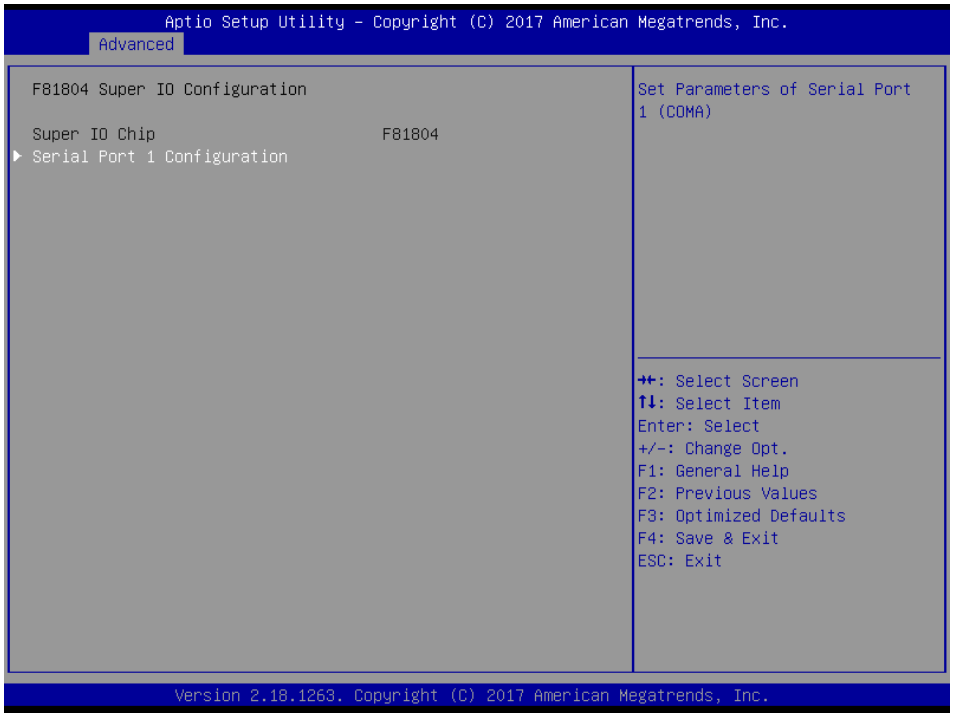
Step 1. 进入 BIOS，启动主机电源，立即按住 < DEL > 键。

Step 2. 随后会进入 BIOS 配置页面，您可从 BIOS 配置列表进入其它设定页面，如 Advanced 和 Chipset 清单等。

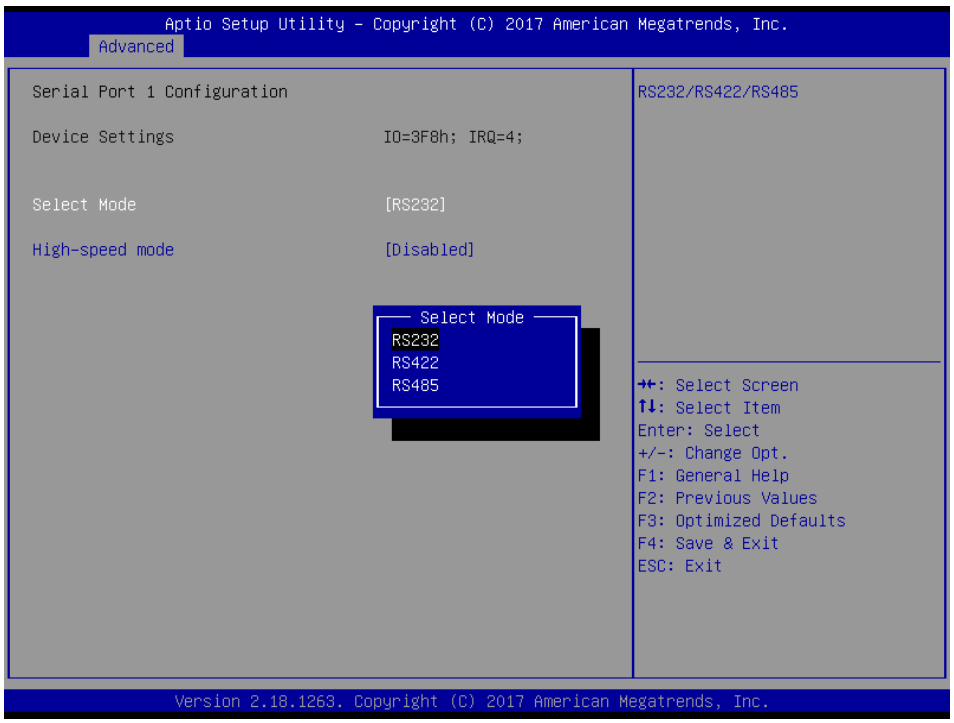
Step 3. 使用键盘←→键进入 Advanced 功能页后，单击【F81804 Super IO Configuration】功能。



Step 4. 使用键盘↑↓键，选择【Serial Port 1 Configuration】可设定 COM 的类型。



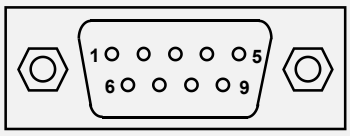
Step 5. 选取【Select Mode】以设定 COM 类型，可设定为 RS232、RS422、RS485，出厂值为 RS232。



Step 6. 设定完成后，需按 **F4** 键保存&离开，才能完成设定。

1

1.5.5 DIO 引脚定义（DB9）



脚位	讯号
1	DIO 0
2	DIO 1
3	DIO 2
4	DIO 3
5	DIO 4
6	DIO 5
7	DIO 6
8	DIO 7
9	GND

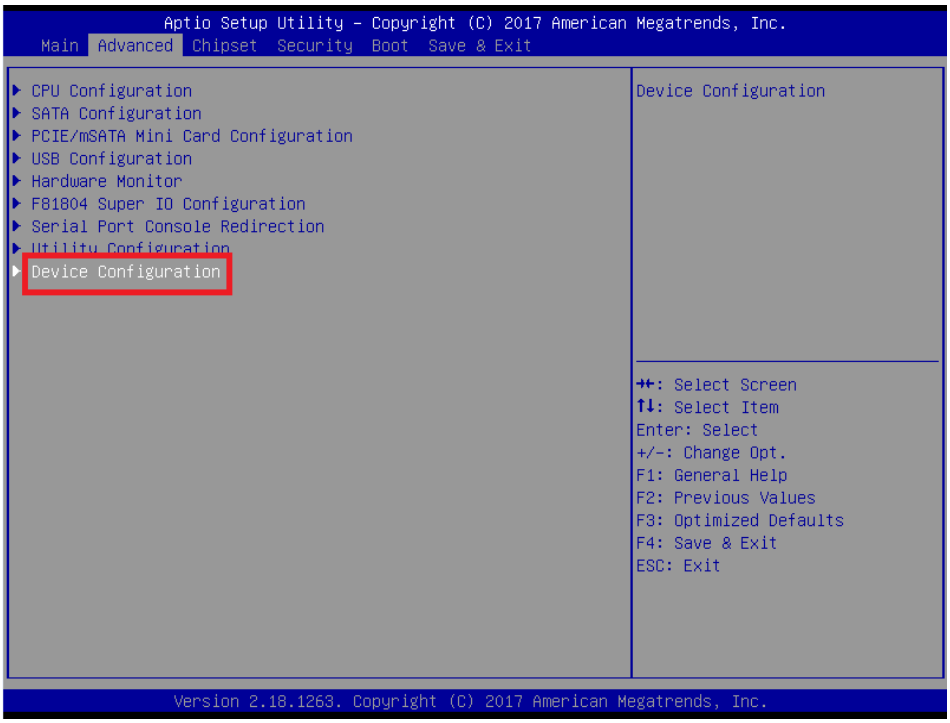
1.5.6 设定 DIO 状态

DIO 出厂值 pin 1~4 为 Out，pin 5~8 为 In，若要变更 DIO 参数，可参考以下操作步骤：

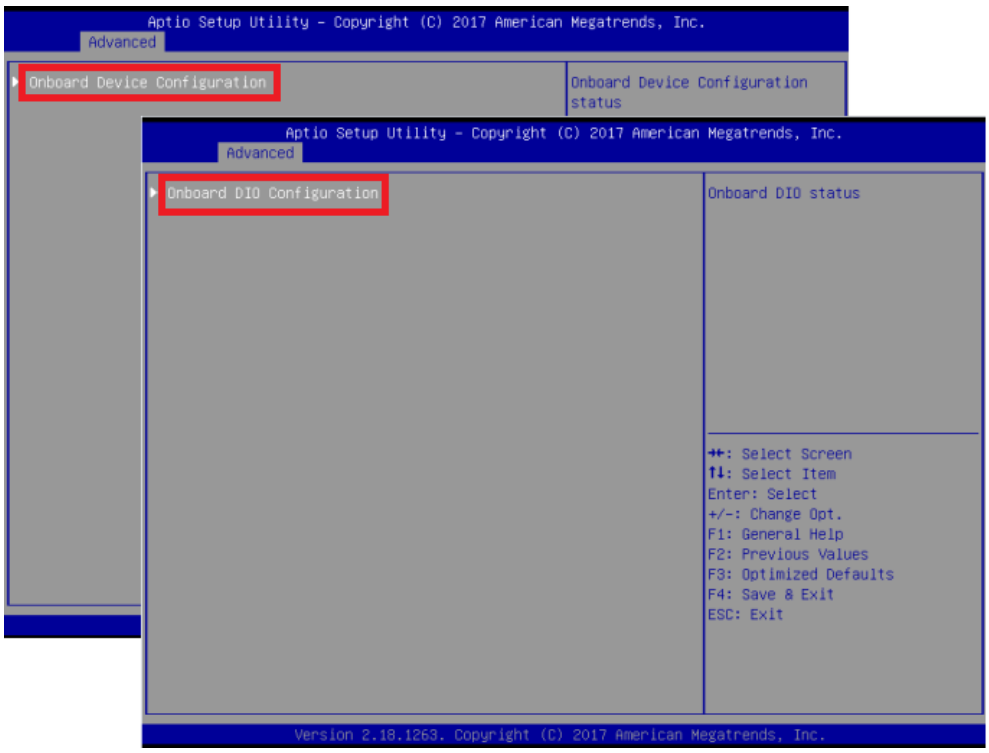
Step 1. 进入 BIOS，启动主机电源，立即按住 < DEL > 键。

Step 2. 随后会进入 BIOS 配置页面，您可从 BIOS 配置列表进入其它设定页面，如 Advanced 和 Chipset 清单等。

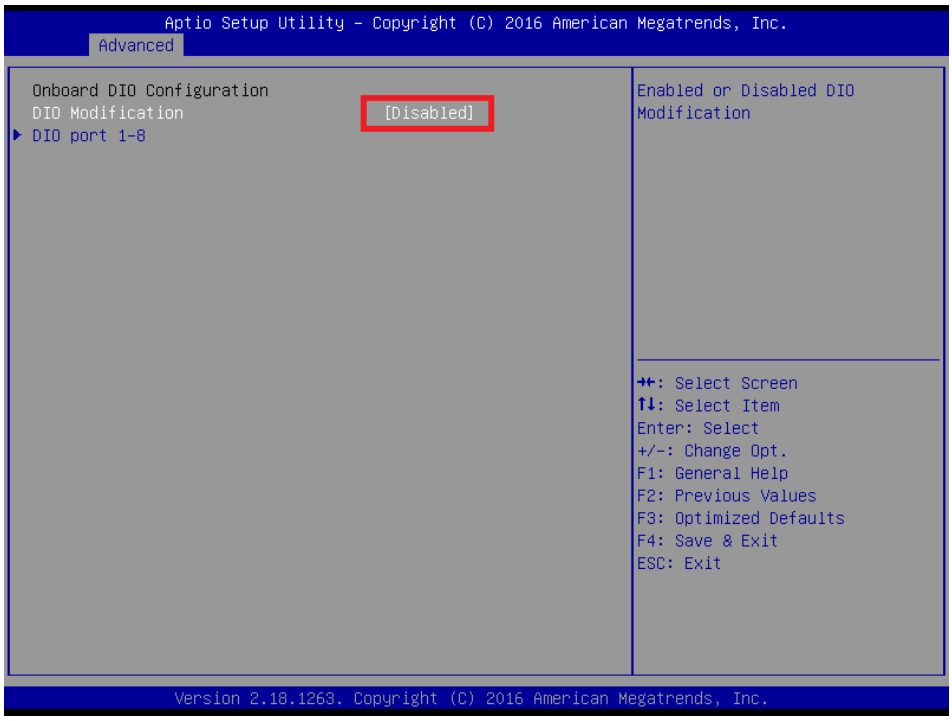
Step 3. 使用键盘←→键进入 Advanced 功能页，选择【Device Configuration】功能。



Step 4. 选择【Onboard Device Configuration】→【Onboard DIO Configuration】。

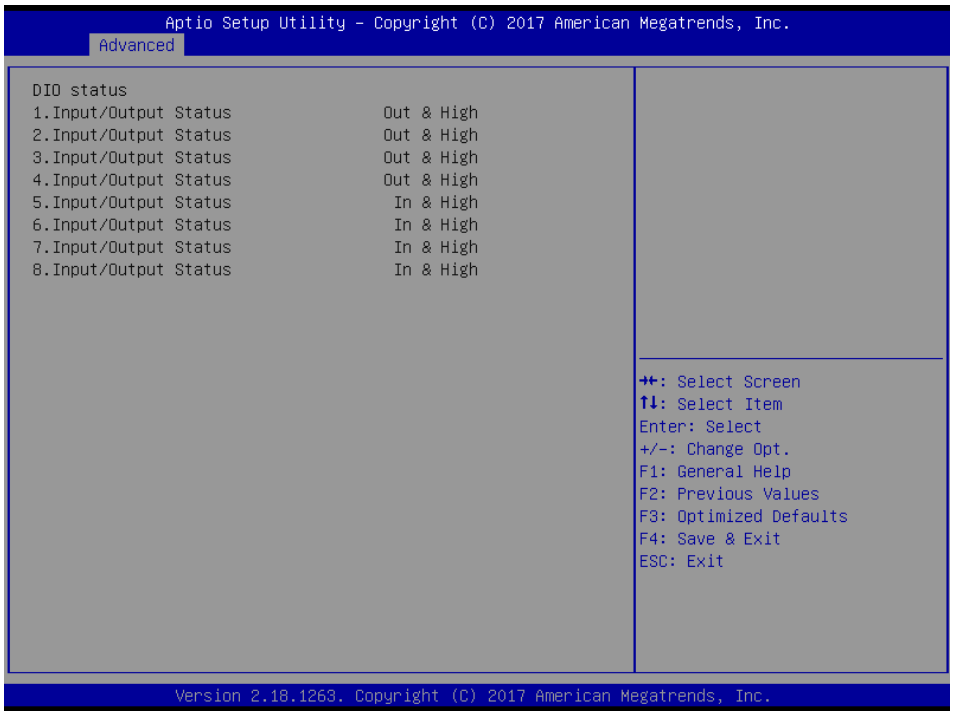


Step 5. 每次开机【DIO Modification】都是 Disabled 的状态，不代表功能关闭，无须紧张。

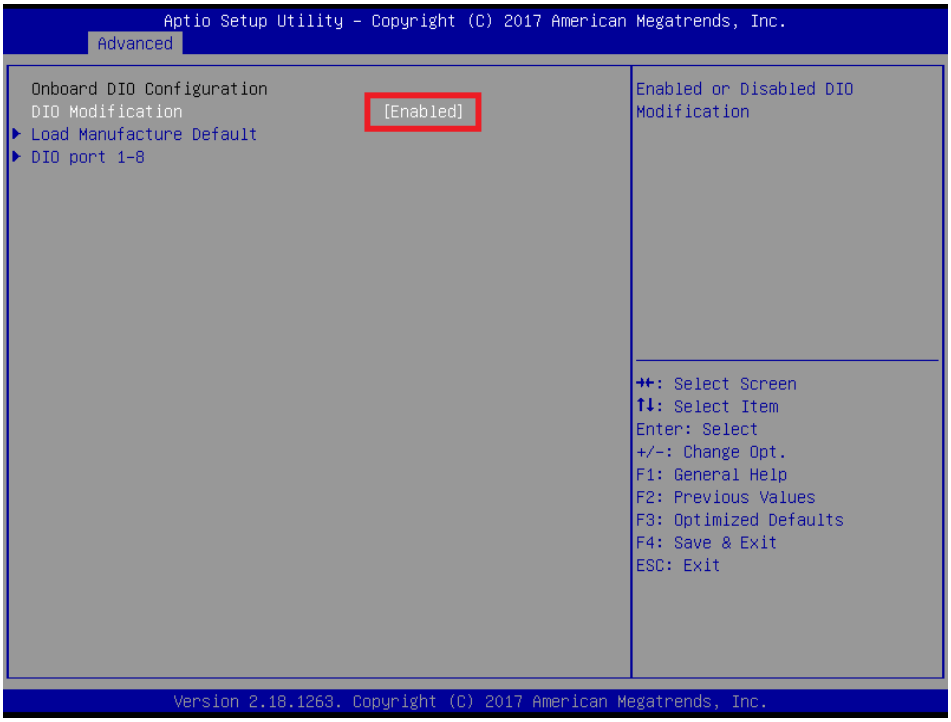


1

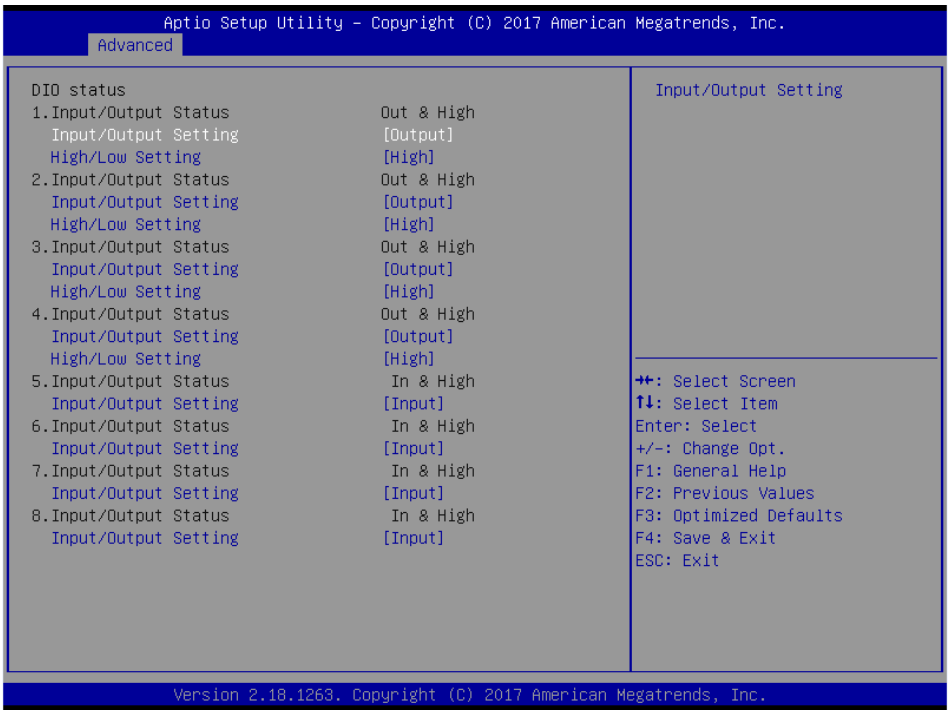
Step 6. 使用键盘↑↓键，在前一步骤页面单击【DIO port 1-8】，可检视 DIO 各 pin 脚的状态，下方为出厂默认状态。



Step 7. 若要修改 DIO 的配置，先将【DIO Modification】切换成 Enabled。



Step 8. 单击【DIO port 1-8】，可设定 pin 脚的状态。



Step 9. 设定完成后，需按 **F4** 键保存&离开，才能完成设定。

MEMO

2

第2章　开始使用 DIALink

目录

2.1　登入系统 2-3
2.2　设备总览 2-4
2.2.1　CNC 信息看板 2-6
2.2.2　新增设备 2-8
2.2.3　点位设定 2-17
2.2.4　触发点位 2-30
2.2.5　刀具管理 2-33
2.2.6　维护管理 2-36
2.2.7　加工程序 2-38
2.2.8　参数补偿 2-39
2.2.9　历史警报 2-39
2.2.10　设备设定 2-40
2.3　排程触发 2-41
2.4　事件与动作 2-42
2.5　查询功能 2-49
2.6　警报清单 2-49
2.7　服务状态 2-50
2.8　系统配置 2-51
2.8.1　一般设定 2-52
2.8.2　通知设定 2-53
2.8.3　网络设定 2-54
2.8.4　排班管理 2-54
2.8.5　Line 设定 2-56
2.8.6　WeChat 设定 2-56
2.8.7　E-mail 设定 2-57

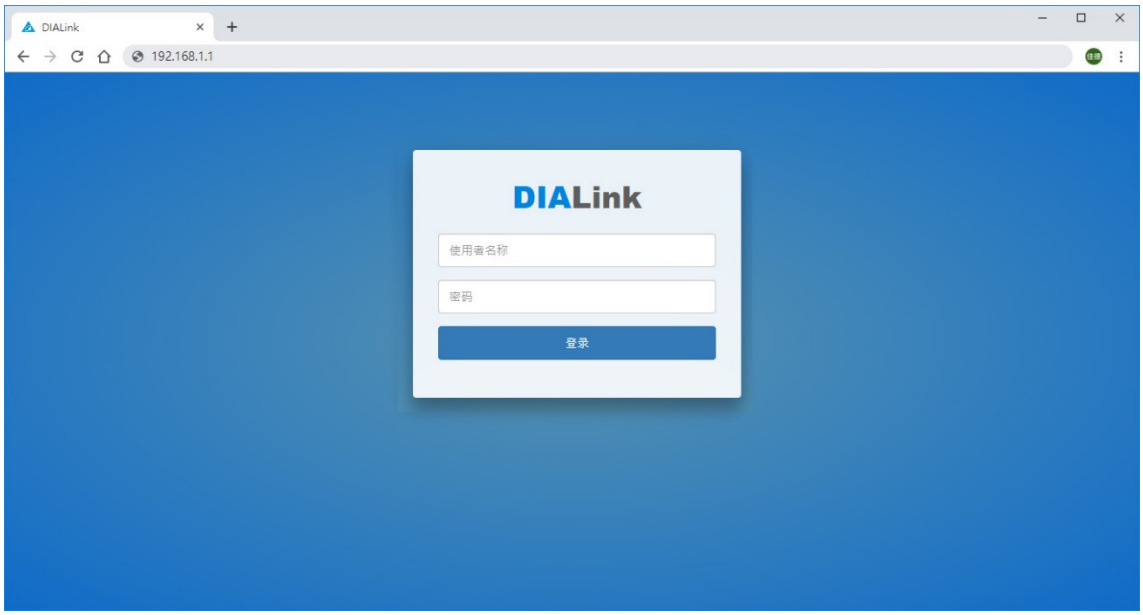
2.8.8 Modbus Slave 设定 2-58

2.1 登入系统

使用时，务必将配件盒里的 USB 加密锁插到主机的任一 USB 孔，DIALink 才能启动对应的设备联机授权。DIALink 为标准的 Html5 网页平台架构，可通过浏览器（Chrome，Firefox 或 IE11 以上版本）进行设置与看板检视。

网卡 A 的默认 IP 为 192.168.1.1，网卡 B 的默认 IP 为 192.168.2.1。可通过 Windows 内的【控制面板】>【网络和因特网】>【网络联机】或 DIALink 内的【系统配置】>【网络】修改网卡 IP。

在浏览器的网址栏输入 DIALink 本机的 IP 地址（http://localhost:5000），进入 DIALink 的登陆页面，用户名默认为 root，密码默认为 admin。登陆后，单击 DIALink 内的【系统配置】>【一般】>【用户设定】来修改用户名与密码。



2.2 设备总览

用户成功登入系统后，页面将转到 DIALink 操作界面，系统首页为设备总览，显示本机通过网络 IP Address 或序列 Serial 接口相互连结的设备名称，已联机的设备显示为绿色（在线），未联机的设备显示为红色（脱机）。点选各设备区块，进入该设备的点位、触发点位、维护管理、看板（看板以后的页面只有 CNC 设备可看到）、设备、刀具管理、维护管理、加工程序、参数补偿、历史警报以及设置页面。点选 CNC 设备类型，进入默认页面为【看板】页面；点选其他设备类型，进入默认页面为【点位】页面；尚未新增点位的 DIALink，可参考第 2.2.2 节新增设备功能的介绍。



左半部为主要系统功能，共有【设备总览】、【排程触发】、【事件动作】、【查询功能】、【报警清单】、【服务状态】、【系统配置】七大功能，各功能简介如下：

【设备总览】：　用户可分别针对不同设备设置，类型有 CNC、Computing、Modbus、OPCDA、OPCUA、PC、PLC 及 Plastics 等通讯装置，并建立设备点位，亦可设置各点位的事件触发条件，已联机设备显示绿色，未联机设备显示红色。(若有购买 SECS 模块授权，则可新增 SECS 设备，此类型设备显示的功能页面跟其他设备会完全不同)

【排程触发】：　用户可设置固定频率或特定时间藉由 E-Mail、SMS、LINE、WeChat、WriteTag 或 Webhook 定时（检查点位数据条件）作提醒通知。

【查询功能】：　显示设备状态维持时间。

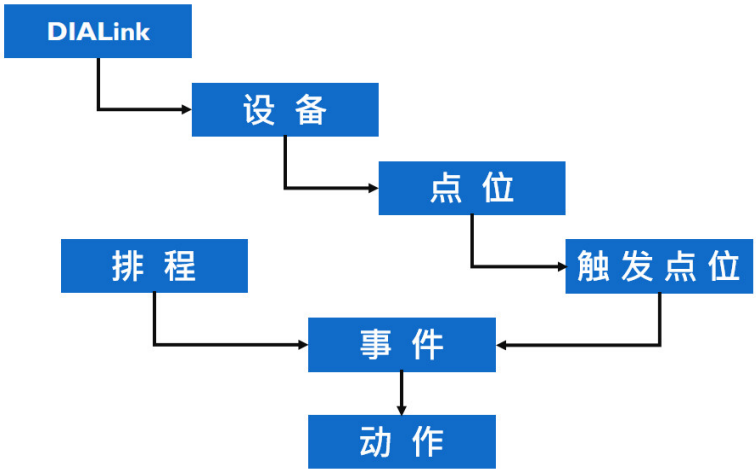
【事件动作】：　用户可用此功能区块建立 / 编辑系统发送的事件与动作通知。

【报警清单】：　供用户查询设备、装置、点位以及系统相关的警报信息。

【服务状态】：　显示 DIALink 主机的 CPU、内存与硬盘使用率。

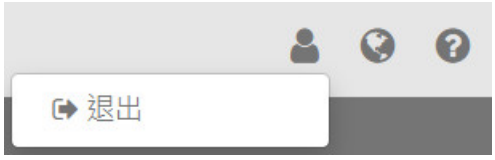
【系统配置】: 设置 MQTT Broke IP、用户帐户、电子邮件与短信主机等相关参数。

若将上述 DIALink 的功能结构图形化，则如下图所示：



主页面右上角的【快捷功能列】有三大功能，各功能说明分别如下：

- **【用户管理】**: 选择，可于此处退出账号。



- **【语系设定】**: 选择，可切换语系，共有 English / 繁体中文 / 简体中文可供选择。



- **【使用手册】**: 单击，选择操作手册，可供用户取得 DIALink 操作手册。



- **【API 文件】**: 单击，选择 API 文件，提供用户查询 API 的在线文档，使用方式参考第 3 章介绍。

- **【DIALink 信息】**：点选 ，选择【关于 DIALink】，显示软件版本、授权类型与数量及期限，以及发行说明。授权数量为设备或点位择一授权，以下图为例，DIALink 可授权最多 10 台设备或最多 1000 点位。发行说明则可下载或在浏览器显示每次的更新说明。



2.2.1 CNC 信息看板

选择 CNC 设备类型（在已新增 CNC 设备与点位的情况下），进入默认页面【看板】页面，此页面显示六大 CNC 信息，分别为三轴信息、CNC 状态、稼动率、警报信息、主轴信息以及运转信息。





【三轴信息】：分为坐标、电流、与制程信息，其中，坐标信息又显示各轴的绝对坐标、机械坐标与相对坐标，剩余坐标则为当前坐标与目标坐标间的差值，坐标与电流信息中的单位在新增默认点位后，利用修改点位的方式新增单位，方便用户更直观地检视。



【稼动率】：在新增默认点位当中的 Information 类别，需加入 Work_Rate，并需事先设置班表，进入班表的时间区间后，稼动率会从 100%开始，只要状态不为 RUNNING，不论是 IDLE 或 ALARM 等都会逐渐减少稼动率，在 CNC 的设备图左边可看到稼动率圆饼图，也可在 CNC 看板画面右边看到更详细的稼动率圆饼图。

例如此班的时间为 14：00～19：00，则 14：00 开始就以下列公式来计算稼动率，跨日班表不会因为过了半夜 12 点影响稼动率计算。

$$\frac{(\text{5 小时- 非 RUNNING 时间})}{\text{班表总时间 5 小时}} = \text{稼动率}$$

【每日机台稼动率】：稼动率（与 Dashboard 圆饼图、班表无关），是设备每日的四种状态的比例：

WorkBusyRate：每日机台运作比率

WorkIdleRate：每日机台闲置比率

WorkAlarmRate：每日机台警报中比率

WorkOffRate：每日机台关机比率

$$Total\ \ Time = Accumulation\ \ of\ \ (busy + alarm + idle + off)\ \ time$$

$$WorkBusyRate = \frac{Accumulation\ \ of\ \ busy\ \ time}{Total\ \ Time}$$

$$WorkAlarmRate = \frac{Accumulation\ \ of\ \ alarm\ \ time}{Total\ \ Time}$$

$$WorkIdleate = \frac{Accumulation\ \ of\ \ idle\ \ time}{Total\ \ Time}$$

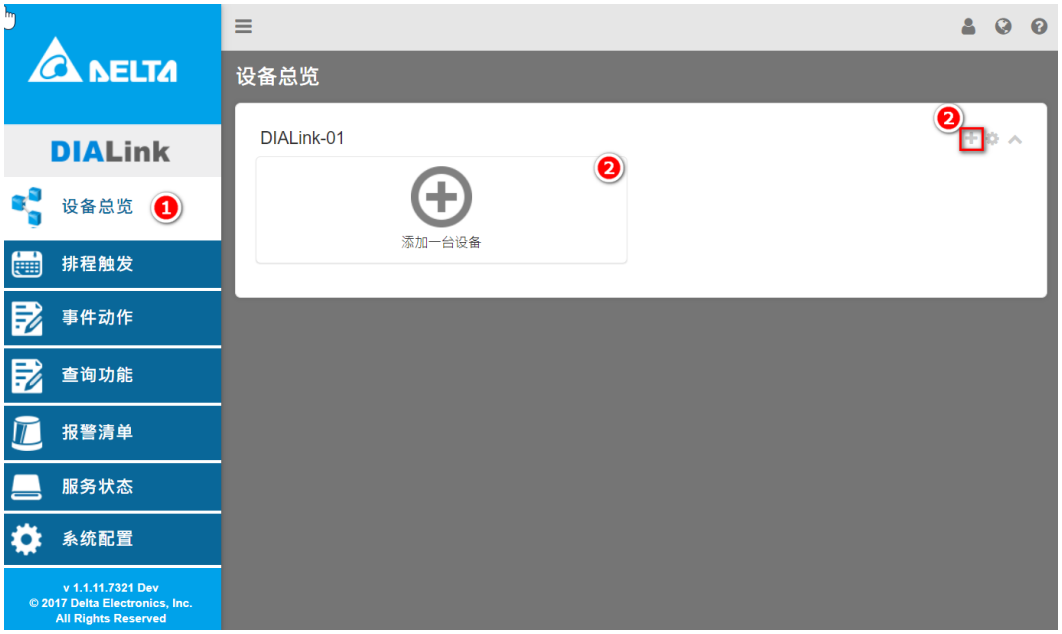
$$WorkOffRate = \frac{Accumulation\ \ of\ \ off\ \ time}{Total\ \ Time}$$

$$WorkBusyRate + WorkAlarmRate + WorkIdleRate + WorkOffRate = 100\%$$

【注】以上四种 WorkRate 为一天内时间的累计，依设备当下状态（Status code）发生所累积的时间，四种状态累计时间于每天零时零分自动清空为 0，并重新计算。

2.2.2 新增设备

首次建立设备的 DIALink，依次单击 ❶左方【设备总览】 > ❷【新增一台设备】，若已新建设备，单击右上方➕符号。





页面右方显示设备相关参数设置，分为 CNC、Computing、Modbus、OPCDA、OPCUA、PC、PLC 及 Plastics 等多种设备类型，根据不同设备类型，系统会带出相应的参数，用户根据需求自行选择 / 输入设备相关参数。目前有支持的各大主流厂牌，如下表：

类型	各大主流厂牌/型号	
CNC	1. CNC Simulator - Modbus TCP Protocol 2. DELTA -NC300 3. FANUC -0i/16i/18i/21i/30i/31i/32i 4. HEIDENHAIN -iTNC530 -iTNC640	5. MITSUBISHI -M70/M700 Series -M80 Series 6. SIEMENS -840D 7. Syntec -Syntec

类型	各大主流厂牌/型号	
Computing	Edge Computing	
Modbus	Modbus	
PC	DELTA	DELTA
PLC	1. DELTA -Delta 15MC TCP（Ethernet） -Delta AH Series（COM Port） -Delta AH Series PLC（Modbus TCP）（Ethernet） -Delta AS Series（COM Port） -Delta AS Series PLC（Modbus TCP）（Ethernet） -Delta DVP Series（COM Port） -Delta DVP TCP/IP（Ethernet） -Delta RTU-EN01（Ethernet） 2. Keyence -KV Series TCP（Ethernet） 3. Mitsubishi -FX3U Ethernet（Ethernet） -Q Series Ethernet（Ethernet） -Q Series Ethernet 3E（Ethernet）	4. Omron -C Series TCP （Ethernet） -CJ2M TCP（Ethernet） -CP1 Series（COM Port） -CP1 Series TCP（Ethernet） -NJ Series FINS TCP（Ethernet） 5. Panasonic -FP Series TCP（Ethernet） -FP7 Series TCP（Ethernet） 6. SIEMENS -S7 1200（Ethernet） -S7 1500（Ethernet） -S7 200（Ethernet） -S7 200 Smart（Ethernet）
Plastics	Plastics -OPC -PLC	

备注：请务必确认 CNC 控制器的数据采集授权（DNC 授权）已经开通，否则可能会无法联机或连上但无法取得数据。

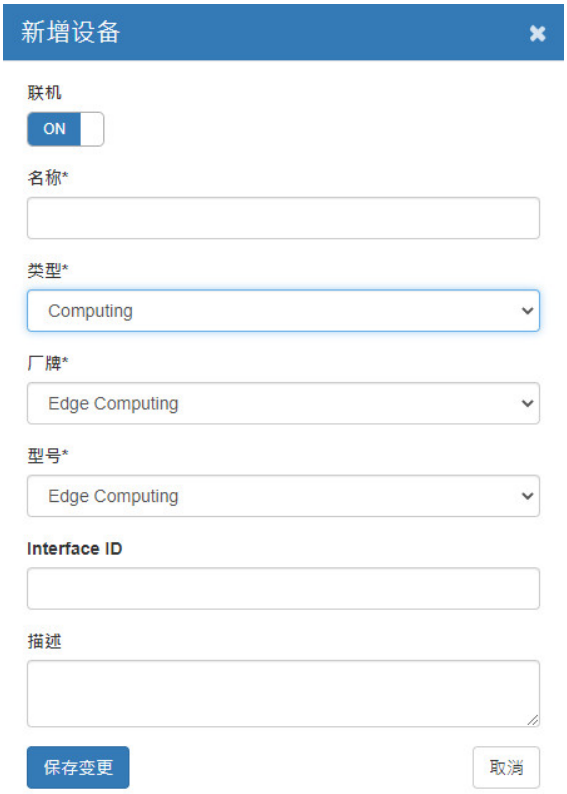
星号（ * ）为必填项目，输入完毕后，单击下方【保存变更】按钮，若输入值不符合系统规则，表格边框会变为红线，提示错误信息。

设备联机功能可随时被设为关闭或开启，相关参数的新增或编辑皆须单击参数页面下方的【保存变更】按钮。

- **建立 CNC 装置页面**



- **建立 Edge Computing 装置页面**



- **建立 Modbus 装置页面**



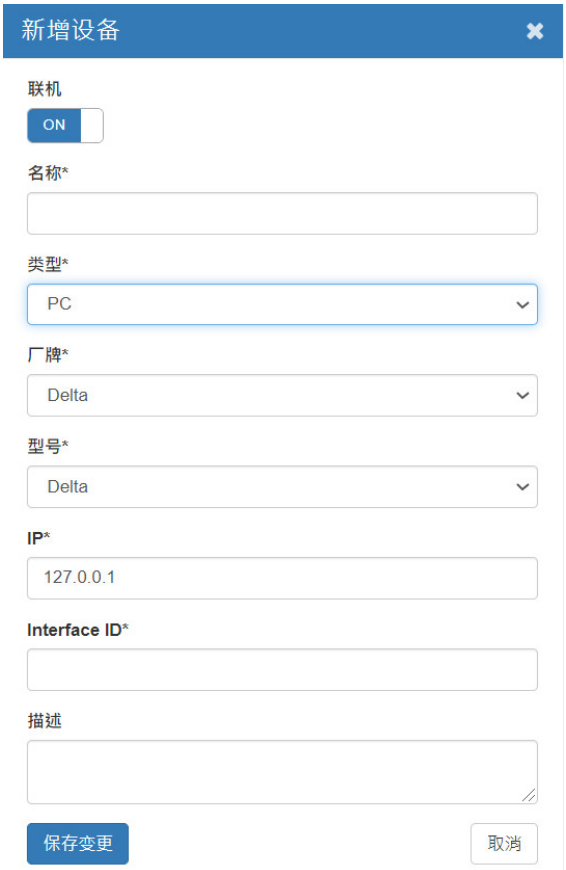
- **建立 OPCDA 装置页面**



- **建立 OPCUA 装置页面**



- **建立 PC 装置页面**



目前 DIALink 只支持 Delta 工厂的 PC 自动化设备，透过 Http 通讯主动上传特定格式的数据内容。

- **建立 PLC 装置页面**



- **建立 Plastics 装置页面**



其中，OPC UA（OPC Unified Architecture）标准通讯协议可与服务器进行联机及数据读、写。

DIALink OPC UA Client（客户端）联机支持设置如下：

项 目	说 明
IP	OPC UA 服务器 IP
通讯端口	OPC UA 服务器通讯端口
URL	OPC UA 服务器端点 URL
服务器名称	OPC UA 服务器所有端点名称 安全策略： None: 无安全 Basic128Rsa15：128 bit 加密 Basic256：256 bit 加密 Basic256Sha256：256 bit 加密 安全模式：

项 目	说 明
	None：无安全 Sign：数据签名 Sign & Encrypt：数据签名并加密 选择安全策略及模式对服务器进行联机： 客户端默认使用自签证书 客户端自动信任服务器端证书 服务器需将客户端证书设置为信任证书
证书	匿名：允许匿名联机 用户：用户名称及密码联机 证书：使用证书检查登入身份
Browse Retry	浏览 OPC UA 服务器 Node 等待次数
Browse Timeout	浏览 OPC UA 服务器 Node 等待时间（单位：毫秒）

OPCUA 变量类型节点内容值读写支持功能如下：

项 目	说 明
支持变量数据类型	SByte, Byte, Int16, UInt16, Int32, UInt32, Int64, UInt64, Float, Double, String, Boolean, Date Time
支持变量类型一维数组	支持数据类型：上述变数数据类型
特殊数据类型以字符串处理	Boolean、Date Time、一维数组数据类型： 对外数据串接（Web API, MQTT, Modbus Slave, OPC UA Server）时，数据类型统一视为字符串 一维数组内容值输出字符串范例： 数值数组：{1,2,3,4} 布尔数组：{True,False,True,False} Date Time 数组： {2020-11-01T16:35:59.5848704Z,2020-11-02T16:35:59.5848704Z,2020-11-03T16:35:59.5848704Z} 字符串数组：{ABCD,EFGH,123456}

设备新增完毕后，选择设备节点右上方 > 【设备】分页，可于第一栏查看设备联机状态，状态图标说明如下表（备注：类型为 Edge Computing 及 PC 的设备由于点位特性跟其他设备不同，没有「断线」状态。Edge Computing 是计算点位，参考其他点位而来，故难定义联机状态。PC 设备则是因为其点位数值是被动等待 Web API 被调用，也难定义，故设备皆无「断线」状态。)



图示	说明
	通讯正常（联机中）
	通讯异常（已断线）
	通讯关闭（停止联机）
	有警报事件

点选子页面，将连结至单一设备看板。若要再次编辑设备通讯参数，可于【设备】分页的动作栏中，单击按钮进行编辑，完成编辑后单击下方【保存变更】即可；欲删除单一设备，可单击动作栏的按钮进行删除。



若要删除多台设备，单击 ❶  按钮进行批次编辑，❷勾选欲删除项目左侧方框，❸单击  按钮进行批次删除。



2.2.3 点位设定

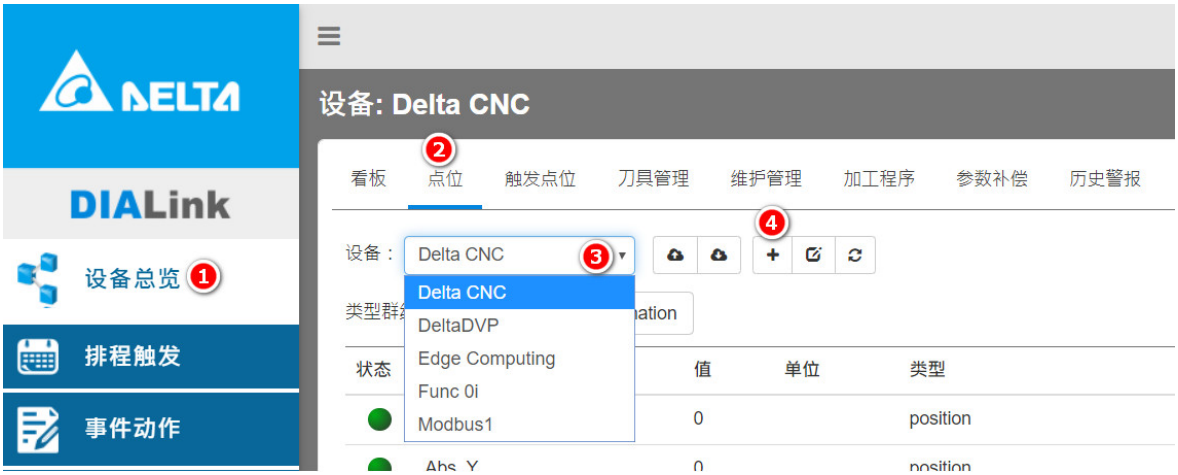
单击❶左方【设备总览】后，选择❷设备节点内已新增的设备以进入点位设定页面，若设备类型为 CNC，会多出看板、刀具管理、维护管理、加工程序、参数补偿、历史警报页面，其余类型显示点位与触发点位页面。



❷于【点位】页面 > ❸切换设备 > ❹单击➕进行新增点位，页面右方显示【新增点位】相关参数设置，默认为单点新增，另一为批次新增，在设备字段处选择已建立的设备名称，根据不同设备类型，系统会带出相应的参数，用户根据需求自行选择 / 输入设备相关参数。

星号（ * ）为必填项目，输入完毕后单击下方【保存变更】按钮；若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。

点位联机可随时被设为关闭或开启，相关参数的新增或编辑皆须单击参数页面下方的【保存变更】按钮。



除 Edge Computing 外，无论是何种设备类型，点位新增分为单点新增与批次新增两种，选择页面上方【单点新增】与【批次新增】可切换新增方式。

类型为 CNC 设备，可切换成默认点位分页，用户可根据需求自行新增 DIALink 内建的 CNC 控制器默认点位。

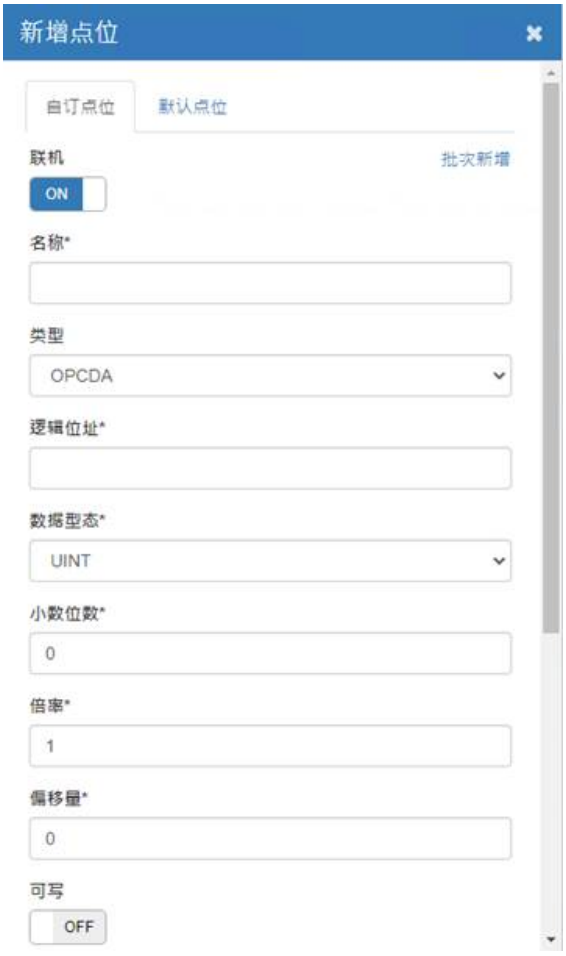
- **单点新增 Modbus 设备点位**



- **批次新增 Modbus 设备点位**



- **单点新增 OPCDA / Plastics 设备点位**



- **批次新增 OPCDA / Plastics 设备点位**



- **单点新增 PLC 设备点位**



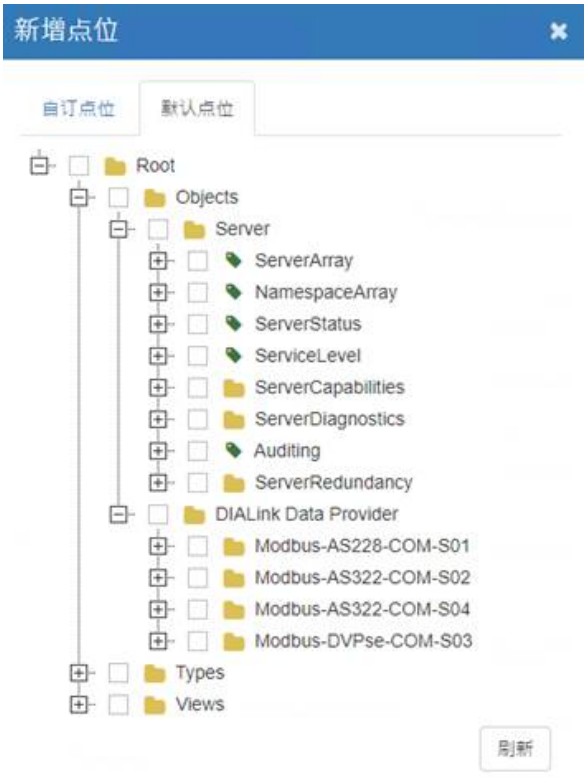
- **批次新增 PLC 设备点位**



- **新增 OPCUA 设备点位**



- **批次新增 OPCUA 设备点位**



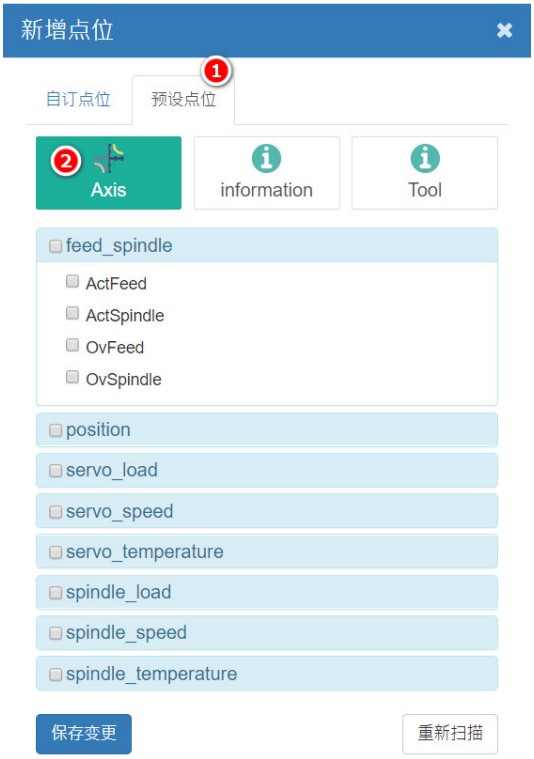
- **单点新增 CNC 设备点位**



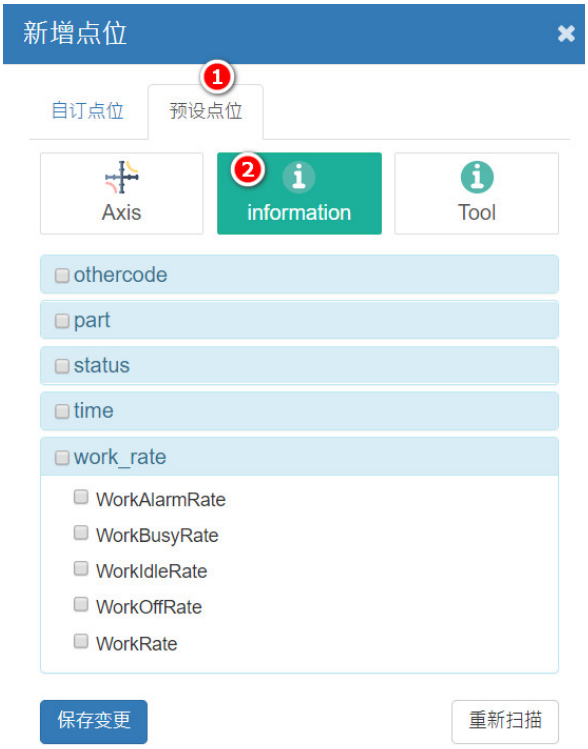
- **批次新增 CNC 设备点位**



- **CNC 装置默认 Axis 点位**

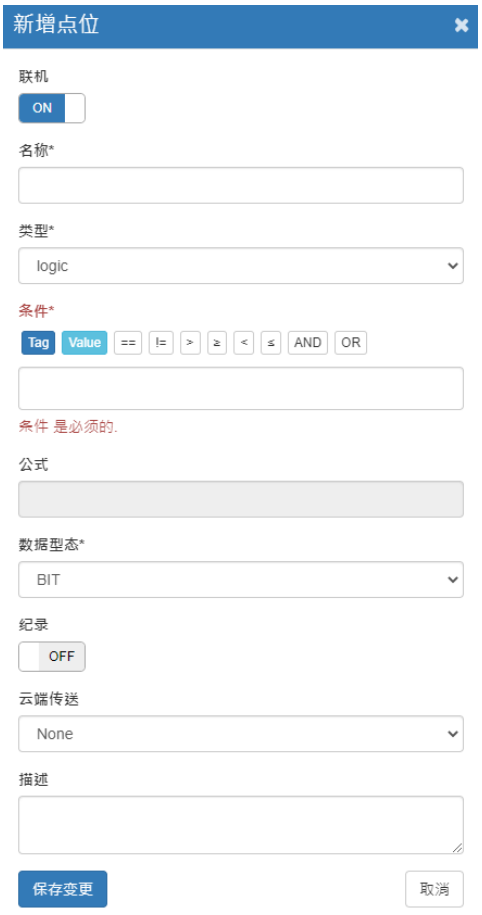


- **CNC 装置默认 Information 点位**

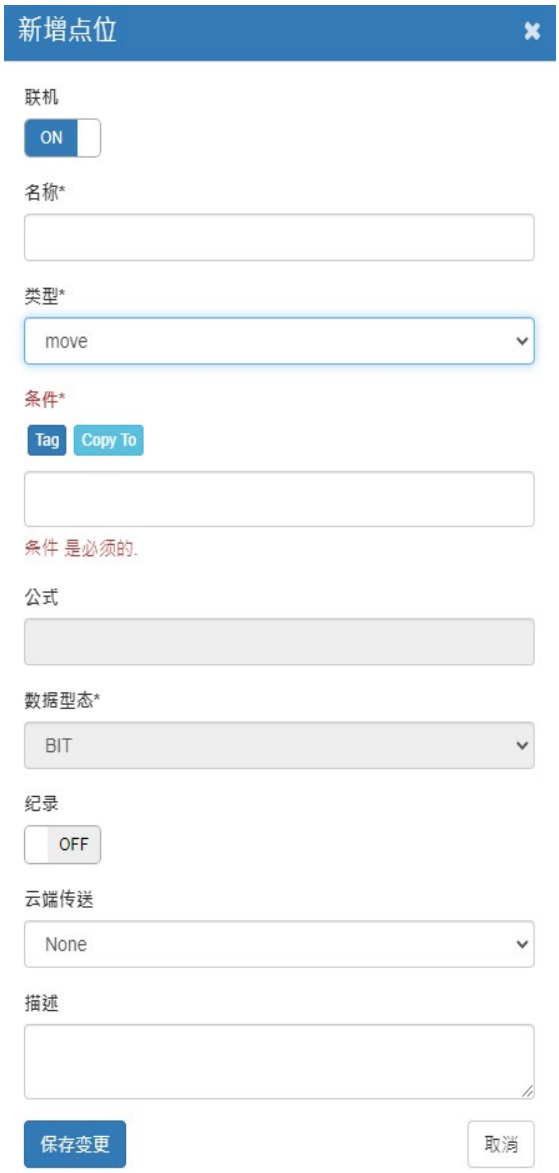


- **Edge Computing 新增虚拟点位（类型：arithmetic、logic、counter、timer、move）**






新增点位
联机
ON
名称*
类型*
counter
条件*
Tag
Value
==
!=
>
≥
<
≤
AND
OR
条件 是必须的.
公式
数据型态*
DOUBLE
小数位数*
0
起始值*
0
递增值*
1
纪录
OFF
云端传送
None
描述
保存变更
取消


新增点位
联机
ON
名称*
类型*
move
条件*
Tag
Copy To
条件 是必须的.
公式
数据型态*
BIT
纪录
OFF
云端传送
None
描述
保存变更
取消



新增 / 编辑虚拟点位，条件字段为必须填写项目，用户可由左向右新增逻辑运算等判断式，额外定义设备点位或常数达到特定条件才进行动作。

- arithmetic 类型

此为点位计算功能，可选取点位与运算符来计算，并不限定多少点位做计算，但是 UI 有文字最大长度限制无法超出。计算结果就会成为点位数据。其中要小心不要在公式当中出现零除的情形，否则该点位数据会变成断线，数据为 NULL。

- logic 类型

此为点位数值比较之功能，目前仅完善支持于两个操作数的比较，三个以上则会有可能与认知不同的判定结果，请勿使用。在数值呈现上会把比较结果为 true 的纪录为 1，false 为 0。

- counter 类型

虚拟点位会记录公式所判断的是 True 的状况一共发生几次，也就是必须要状况为 True 之后，又变为 False，再度变成 True 才会计算一次。

- timer 类型

虚拟点位会记录公式所判断的是 True 的状况一共有多少时间（单位：秒）。

- move 类型

此功能可将既有点位复制到其它点位，但禁止将值复制至虚拟点位。move 也不该设置到无法写入的设备，比方 CNC 的点位或某些 PLC 点位，若设置了不会造成系统无法运行，但是一直有写入错误发生则会略微影响到效能。

图示	说明	运算优先级
Tag	设备点位	0
Value	常数	0
Copy To	复制	1
( )	小括号	1
+	加	3
-	减	3
*	乘	2
/	除	2
==	等于	3
!=	不等于	3
>	大于	2
≥	大于或等于	2
<	小于	2
≤	小于或等于	2
Log10	Log10	2
√	开根号	2
^	次方	2
Sin Cos Tan Cot Sec Csc	三角函数	2
AND	且	2
OR	或	2

新增条件选择点位方式，单击 Tag 选项，依序选择设备、装置，接着选择欲加入判断式的一个或以上的点位，完成后单击【确定】键。

点位新增完毕后，单击【设备区块】中的【点位】分页，可于以下图表第一栏查看设备联机状态，第二栏为实时数值，第三栏为点位名称，如下图 CNC 控制器为例。



若要再次编辑点位参数，可于【点位】分页的动作栏，单击 按钮进行编辑，完成编辑后单击下方【保存变更】即可；若删除单一点位，可单击动作栏的 按钮进行删除。



若删除多笔点位，❶先筛选设备，❷单击 按钮进行批次编辑，❸再勾选左方字段方框选择欲删除项目，❹单击 删除 按钮进行批次删除。

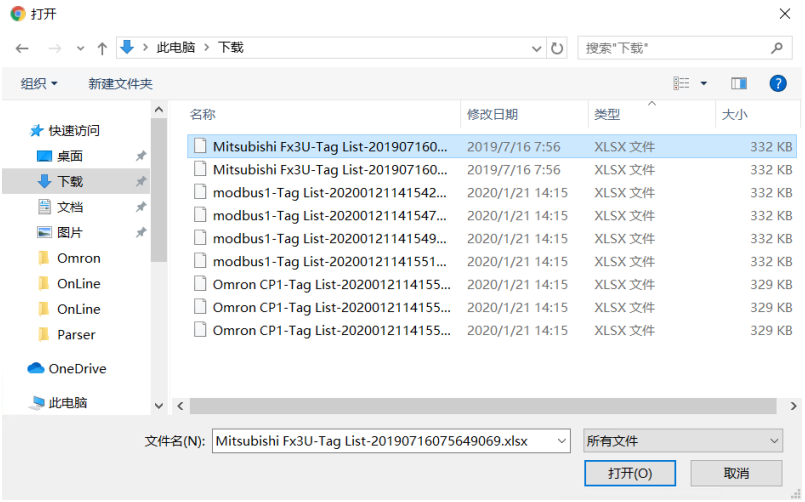


若要导出设备所有点位数据，❶先筛选设备，❷单击  键进行导出。



若要导入设备所有点位数据，❶先筛选设备，❷单击  键选择导入之 Excel 文件，此文件必须是由系统导出之模板。导入文件格式内容说明请参考附录 C。





2.2.4 触发点位

❶于【触发点位】页面 ❷切换设备 > ❸单击 进行新增触发点位，页面右方显示【新增触发点位】相关参数设置，于设备字段处选择已建立的设备名称，系统会带出相应的点位，用户根据需求自行选择 / 输入设备相关参数，星号（ * ）为必填项目，输入完毕单击下方【保存变更】按钮，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。

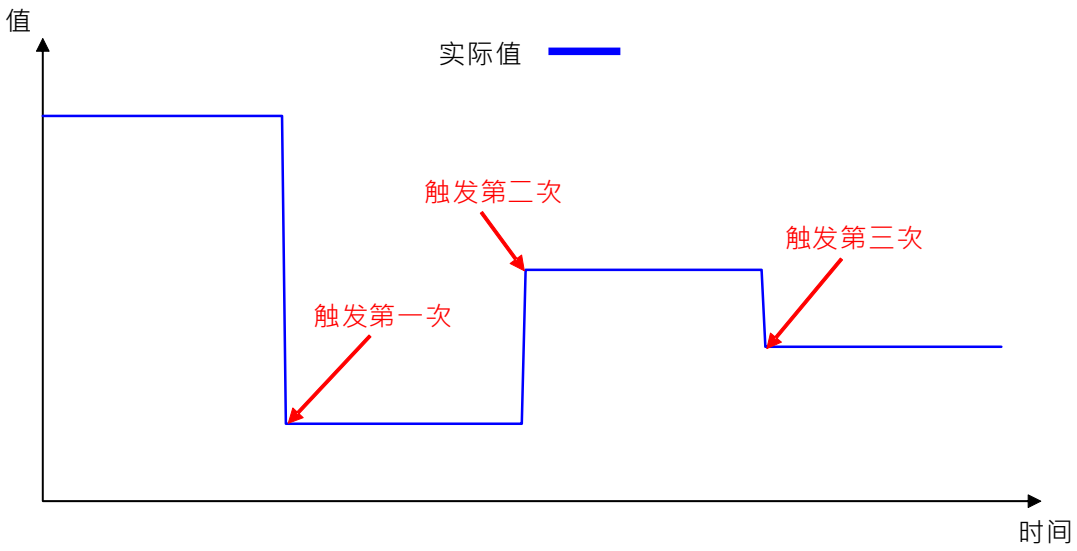


用户可针对不同设备点位的数值，定义额外的触发条件，DIALink 支持的触发类型有五项，分别为 Change、Max、Min、Rising 以及 Falling 五种，各类型触发条件简介如下：

【Change】 ： 数值不等于前一次扫描的数值，触发 Change 条件，一般用于状态变更。

Ex：前次扫描点位数值为 0，下次扫描点位数值为 1，即触发 Change 条件

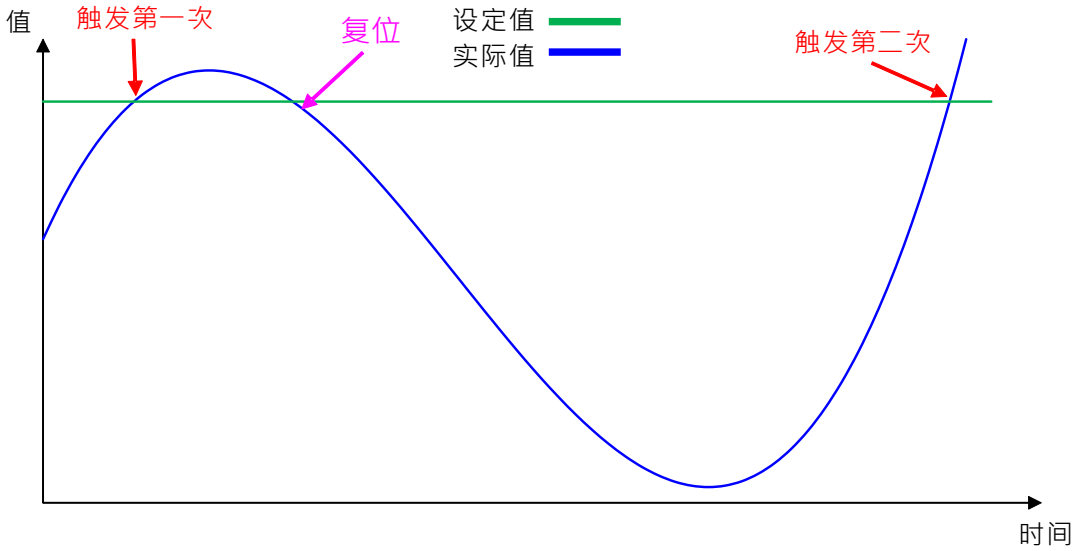
Change 图例：



【 M a x 】 ： 数值大于设定值，触发 Max 条件一次，低于设定值后条件复位；若数值再次大于设定值，触发条件再次启动。

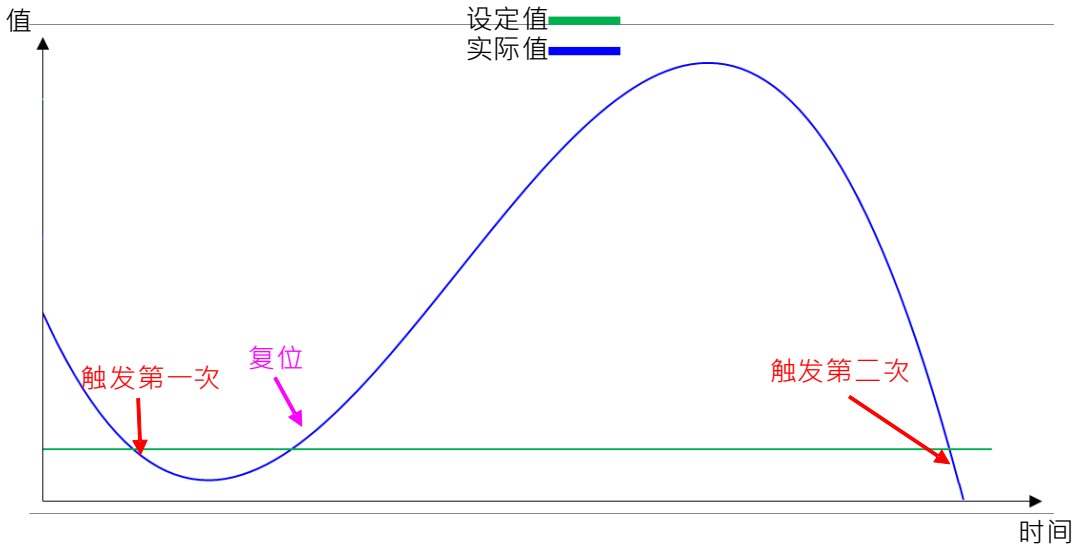
Ex：若 Max 值设为 100，点位数值为 101，即触发 Max 条件，若数值持续大于 100，不再触发 Max 条件，直到数值低于 100 后，数值再次大于设定值时，触发条件才会启动。

Max 图例：



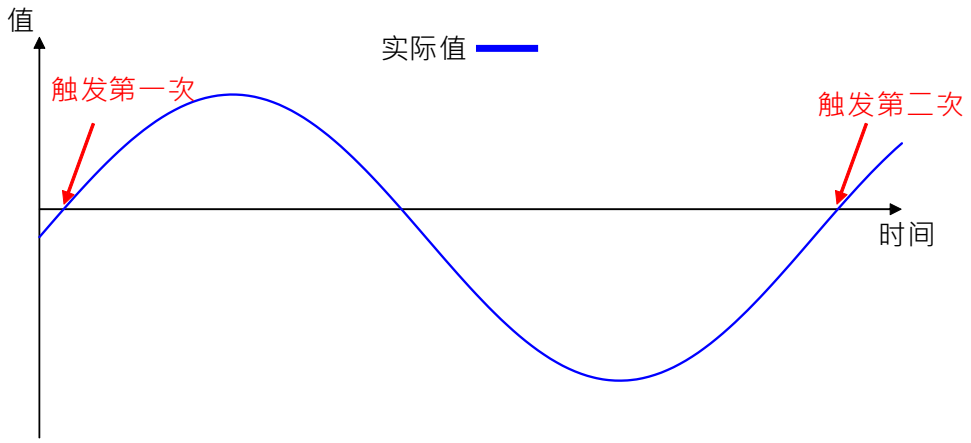
【 M i n 】 ： 数值小于设定值，触发 Min 条件一次，高于设定值后条件复位；若数值再次小于设定值，触发条件再次启动。

Min 图例：



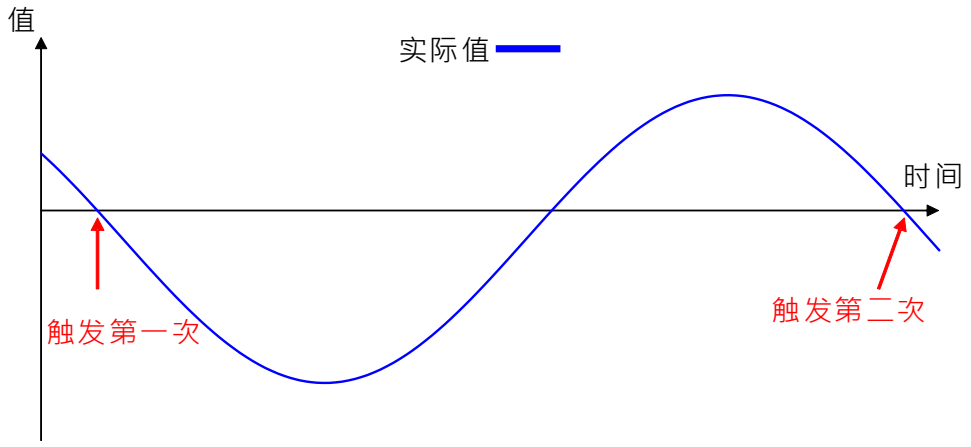
【Rising】 ： 数值大于零，触发 Rising 条件。

Rising 图例：



【Falling】 ： 数值小于零，触发 Falling 条件。

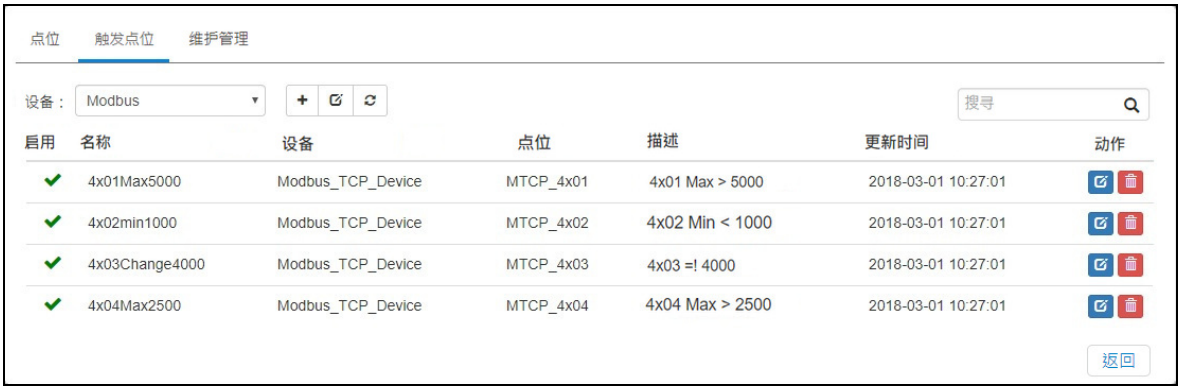
Falling 图例：



- **新增触发点位**



各触发点位需搭配事件才能通报给用户，通报方式有 E-mail、SMS、LINE、WeChat、WriteTag 与 Webhook 六种，可于绑定的事件选择【加入现有事件】或【新增事件】，事件部分请参考第 2.4 节事件与动作介绍，触发点位编辑与批次删除方式参考以上设备与点位小节。



2.2.5 刀具管理

❶于【刀具管理】页面 > ❷切换 CNC 控制器 > ❸单击 新增刀具信息，页面右方显示【新增刀具】相关参数设置，用户依需求自行选择 / 输入相关参数，星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。


DIALink
设备总览
排程触发
设备: Delta CNC
看板
点位
触发点位
刀具管理
维护管理
加工程序
参数补偿
历史警报
1
2
3
设备：
Delta CNC
编号
类型
使用时间(分)
寿命时间(分)
使用次数
寿命次数
进给量
没有结果

- **新增刀具**


新增刀具
编号*
1
类型
寿命时间(分)*
寿命次数*
进给量
主轴转速
工件材质
刀具材质
刀径
刀长
供应商
连络人
电话
描述
保存变更
取消

单击动作字段的 可新增警报动作，根据不同的条件发送 E-mail、SMS、Line 或 WeChat 等动作。其中的 E-mail、Line 以及 WeChat 请在系统配置页面的【E-mail 设定】、【Line 设定】及【WeChat 设定】页签中新增接收者帐户信息，然后才能在下拉选单中选择警报通知的接收者，而 E-mail 的部分可支持发送给多个联络人账号。

条件的部分，可以勾选使用时间是否超过刀具寿命时间，或者使用次数是否超过刀具寿命次数，凡是使用刀具，不论使用多久，都会算一次的使用；也可以设置到达寿命的一定%数就发出警报提醒；可以提前进行保养等维护动作。



- **刀具使用归零**

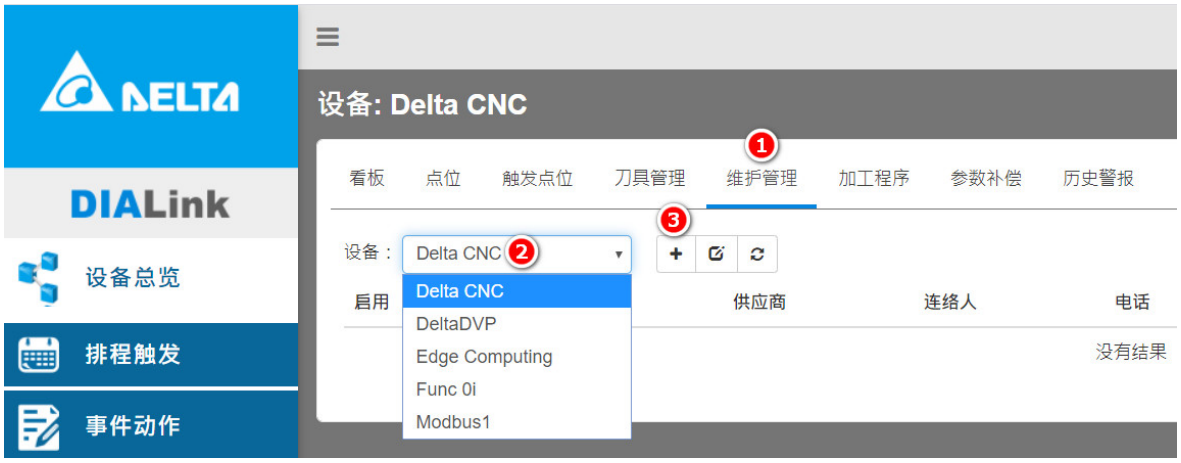
❶于【刀具管理】页面 > ❷在需要归零的刀具列空白处点击 > ❸点击【使用时间归零】按钮，可将该刀具的使用时间（分）归零，点击【使用次数归零】按钮，可将该刀具的使用次数归零。



2.2.6 维护管理

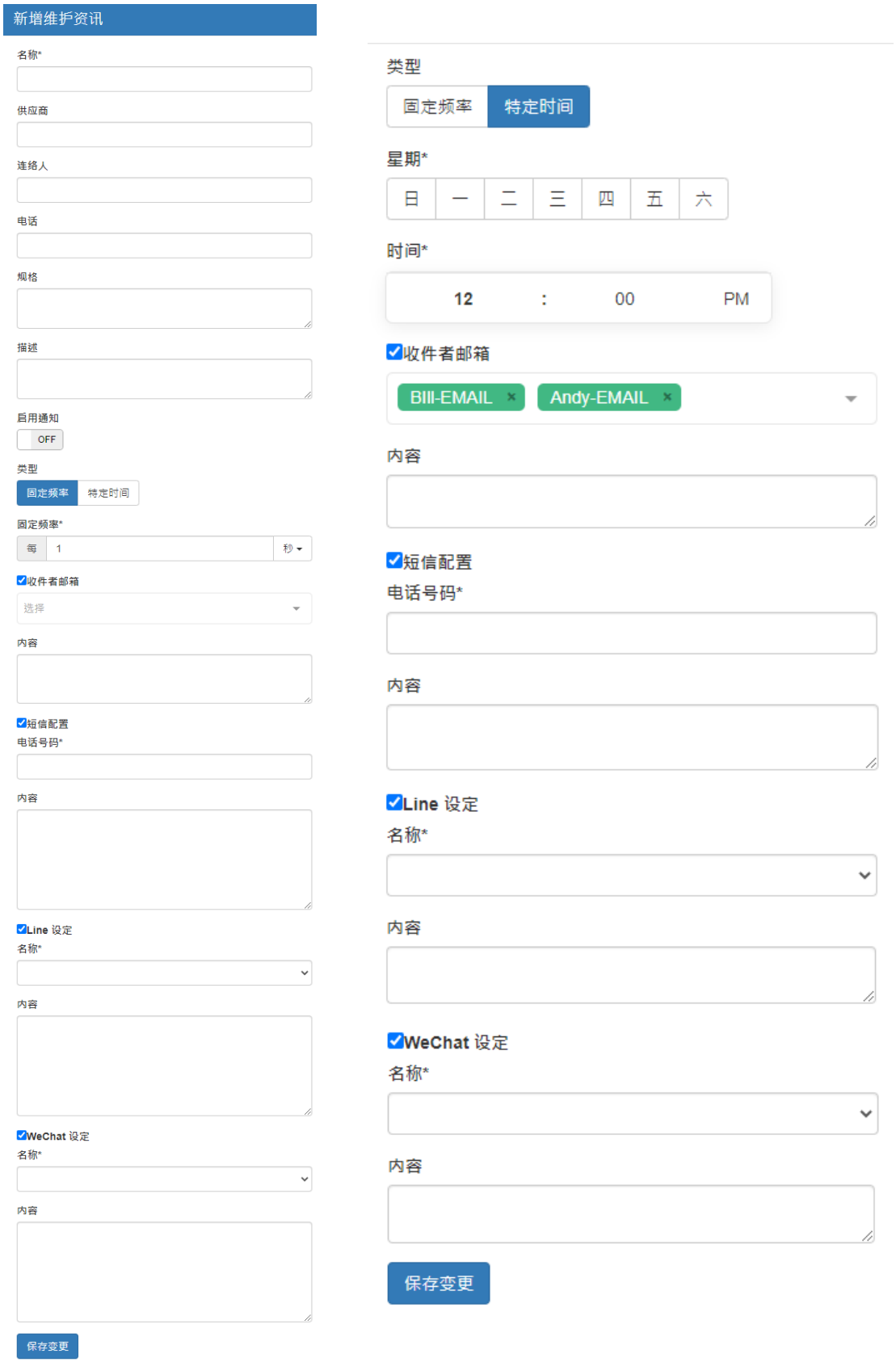
用户可通过维护管理功能，设置固定频率或特定时间以 Email 及短信通知相关人员保养设备，可参考第 2.3 节排程触发介绍的通知频率。

❶于【维护管理】页面 > ❷切换设备 > ❸单击 ➕ 新增维护信息，页面右方显示【新增维护信息】相关参数设置，用户依需求自行选择 / 输入相关参数，星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。



- **新增维护信息：**可自行输入各设备零件的联络信息，并设置通知类型: 固定频率 (秒/时/天) 及特定时间(周一~周日/时间点) 两种类型，固定频率可设置固定多少秒或分钟、或小时或天来发出通

知，而特定时间则是设置接下来礼拜几何时会发出通知。通知的方式可使用 E-Mail 通知（允许通知多人），或者使用 Line 和 WeChat 通知。

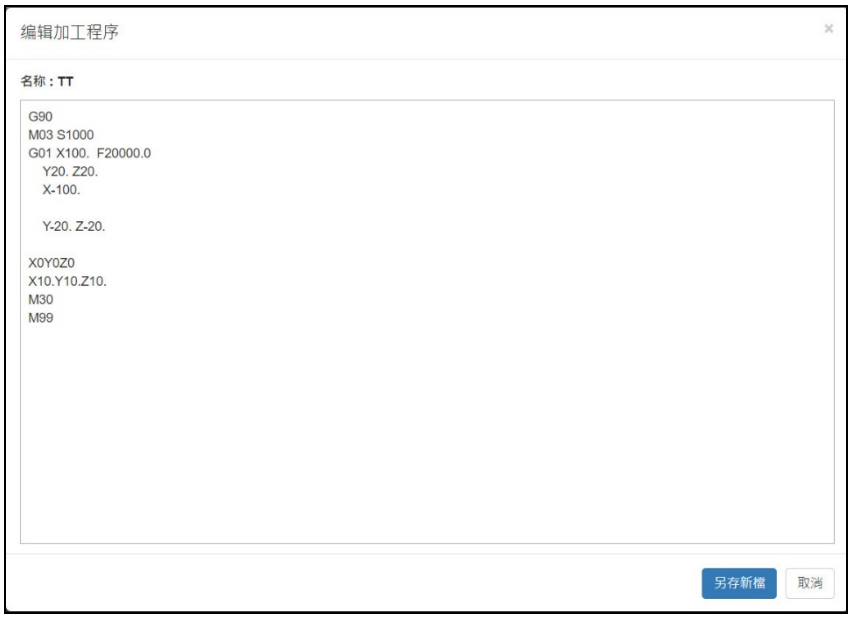


2.2.7 加工程序

DIALink 确实与 CNC 控制器联机后，于【加工程序】页面显示控制器 NC 码文件列表，可将加工程序上传（控制器传到计算机）、下载（计算机传到控制器）、编辑以及删除。



单击可在计算机端编辑加工程序，单击【确认修改】，会直接由 PC 端下载到控制器；单击【另存新檔】，会将文件上传至 PC 端。



如果 CNC 控制器有额外加装保存装置，则有可能无法取得外加保存装置的加工程序文件，目前已经确认 FANUC 是无法读写，而 DELTA 的 CNC 则可以读写。

另外，CNC 在加工运行中透过 PC 端读取加工程序有很高机率会失败。因为加工非常占用资源，文件的处理则是优先级较低，请尽量避免在加工中读写加工程序。

2.2.8 参数补偿

参数补偿页面可显示 CNC 控制器内刀具管理的补偿参数，并在 DIALink 内设置。

❶于【参数补偿】页面 > ❷切换 CNC 控制器，显示控制器内刀具补偿的参数，修改完后点击【写入参数】会做一次性的修改。



Id	Length	Life	Radius	WearLength	WearRadius
1	3.4	3	2.2	1.1	5.5
2	1.2	2	3.2	4.2	5.2
3	-1.3	-3	-3.3	-4.3	-5.3
4	-1.4	-3	-3.4	-4.4	-5.4
5	5	0	0	0	0

2.2.9 历史警报

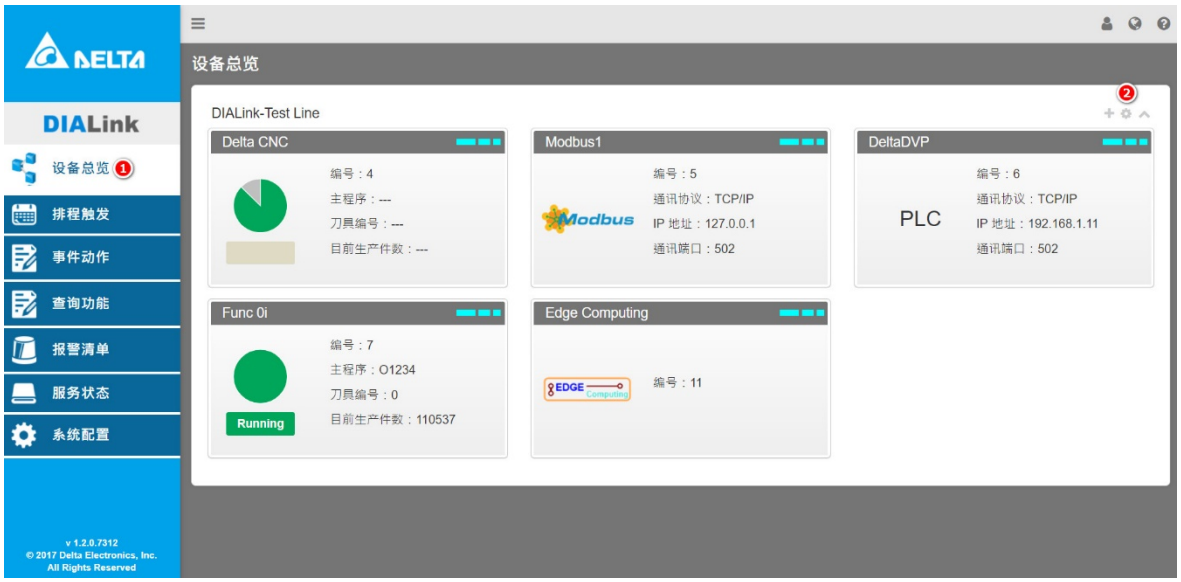
❶于【历史警报】页面 > ❷切换 CNC 控制器，以列表方式呈现给用户关于设备之历史警报信息，每页最多显示二十笔。



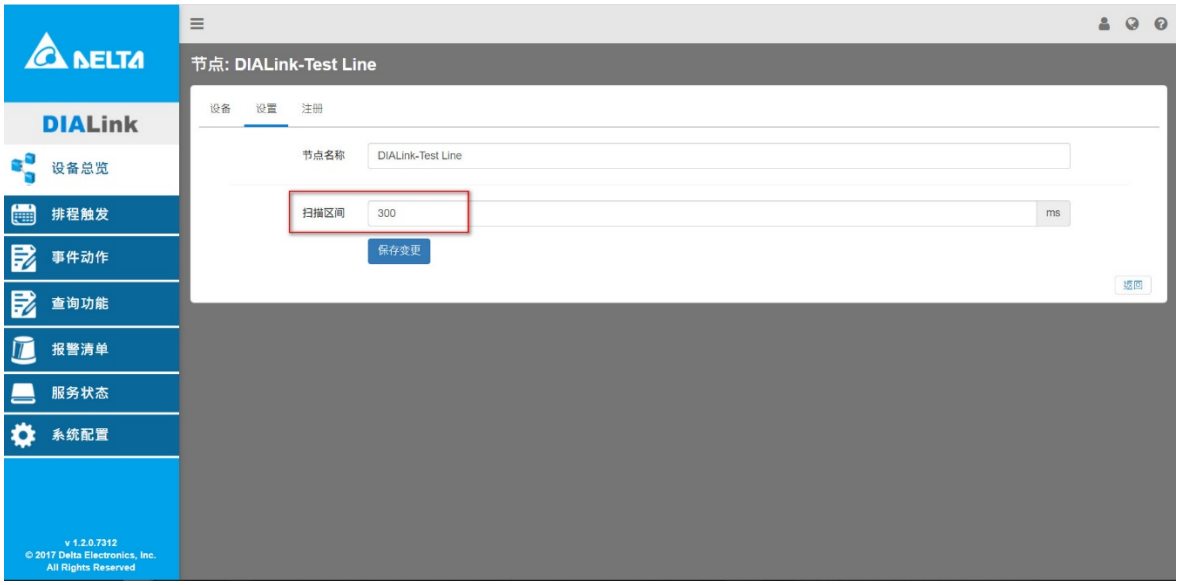
发生时间	消息
2019-12-19 21:57:42	User defined Alarm
2019-12-19 21:57:26	EXECUTE HOME RETURN
2019-12-19 21:57:18	User defined Alarm
2019-12-19 21:57:00	EXECUTE HOME RETURN
2019-12-05 21:50:57	EXECUTE HOME RETURN

2.2.10 设备设定

进入设备设定页面，❶左方【设备总览】功能，❷再单击设备节点右上方⚙。



设备功能区块选择【设置】页面，用户可修改设备名称与设备最低扫描频率，扫描频率默认值为 300ms，代表系统对每一个设备最低轮询取得点位数据的时间。用户最低可设置至 100ms，但因为设备本身效能、网络或串行速度影响着读写效能，有可能设置的扫描区间无法达到。



设备功能区块选择【注册】页面，用户输入有效 PLC 注册码后，才能正确读取 PLC 设备数据。



2.3 排程触发

选择❶左方【排程触发】功能，用户可选择固定频率 / 特定时间以 E-mail、SMS、LINE、WeChat、WriteTag 或 Webhook 方式提醒用户进行自定义动作。

单击❷进行新增排程，页面右方显示【新增排程】相关参数设置，类型分为固定频率与特定时间，于绑定的事件单击【加入现有事件】或【新增事件】选择指定动作，事件部分参考第 2.4 节事件与动作介绍，触发点位编辑与批次删除方式，参考以上设备与点位小节。



【固定频率】：　可输入固定频率触发排程，频率间隔单位可选秒、分钟、小时与天，以新增排程当前时间开始往后计算。

【特定时间】:　可单选或复选每星期几的哪一时间触发排程。

- **新增固定频率排程**



- **新增特定时间排程**



2.4　事件与动作

选择❶左方【事件动作】功能，用户可自由选择是否以指定条件搭配动作以 E-mail、SMS、LINE、WeChat、WriteTag 或 Webhook 方式通报用户，使用 E-Mail 与 SMS 发送功能前，请先至【系统配置】功能 >【通知】页面内输入电子邮件与短信设置参数。使用 Line 与 WeChat 发送功能前，请先至【系统配置】 > 【通知】页面内开启该项功能，并于【Line 设定】与【WeChat 设定】页新增 / 编辑设置参数。

单击❷  进行新增事件，页面右方显示【新增事件】相关参数设置，星号 (*) 为必填项目，输入完毕单击下方【保存变更】按钮。若输入值不符合系统规则，表格边框会变为红线，并提示错误信息，事件编辑与批次删除方式请参考以上设备与点位小节。



- **新增事件与动作**

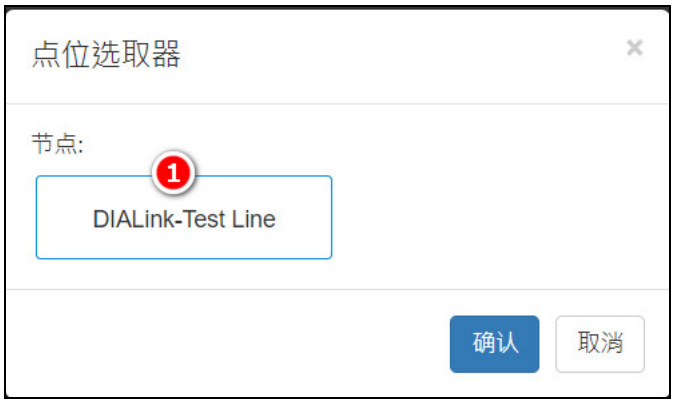


条件字段为非必须填写项目，但用户可由左向右新增逻辑运算等判断式，额外定义设备点位或常数达到特定条件才进行动作。

图示	说明	运算优先级
Tag	设备点位	0
Value	常数	0
==	等于	3
!=	不等于	3
>	大于	2
≥	大于或等于	2
<	小于	2
≤	小于或等于	2
AND	AND 逻辑运算	4
OR	OR 逻辑运算	4
( )	小括号	1

新增条件选择点位方式，单击 Tag 选项，依次选择设备、装置，接着选择欲加入判断式的一个或多个点位，完成后单击【确定】按钮。

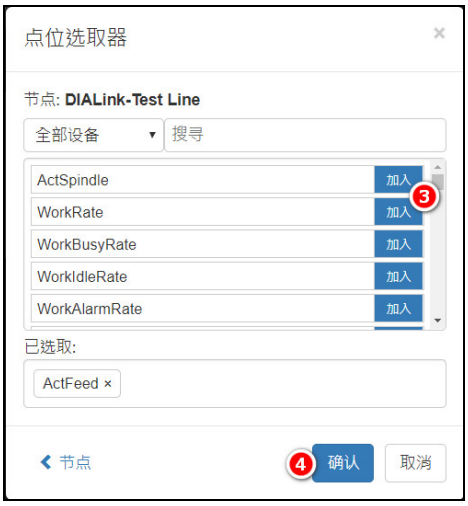
- **事件条件的点位选择器-选取设备**



- **事件条件的点位选择器-选取装置**



- **事件条件的点位选择器-加入点位**



常数输入直接选择 Value 选项，输入数值，完成后单击【确定】按钮。

- **输入事件条件的常数值**



条件字段内的项目无法使用键盘输入 / 删除，鼠标单击可删除项目，左键压住拖曳可改变项目位置，条件选项可为多选移动及删除。完成的条件范例如下图所示，当点位 MRTU_4x01 的数值大于 1000，同时，点位 MTCP_4x02 的数值小于 500 时，允许此事件被触发。

- **条件新增范例**



单击【+新增动作】连结，页面右侧出现动作参数页面，可选择 E-mail、SMS、LINE、WeChat、WriteTag 与 Webhook 六种动作，可选择多种动作，当类型为 E-mail 时，则可选择多个收件者。用户可根据需求设置相关参数，事件与动作新增完毕，分别由动作、新增事件依次单击【保存变更】按钮。

动作内容可带入设备编号（Device ID）与点位编号（Tag ID），语法为{{deviceID|tagID}}，将当前的点位值带入内容中，其中设备编号与点位编号查询方式为分别进入【设备】/【点位】页面，单击已新增的设备 / 点位区块，页面右方会显示设备 / 点位详细信息，即可得知设备 / 点位编号。

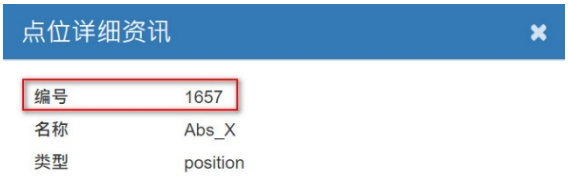
LINE 设定与 Notify 账号申请可参考第 2.8.5 节及附录 B.1。

WeChat 设定与微信公众账号申请可参考第 2.8.6 节及附录 B.2。

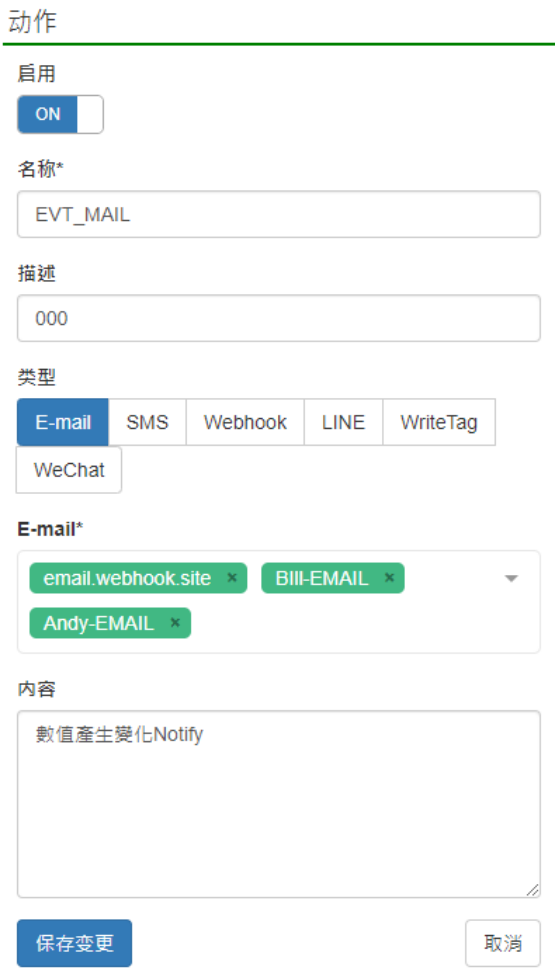
- **设备详细信息**



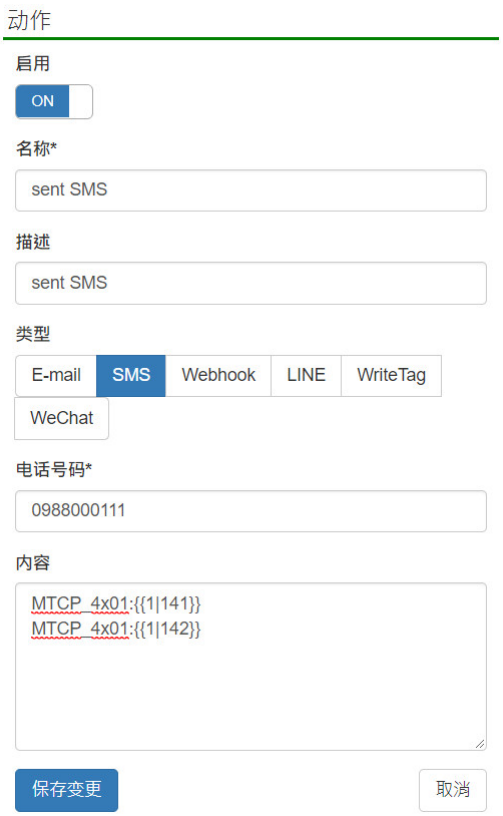
- **点位详细信息**



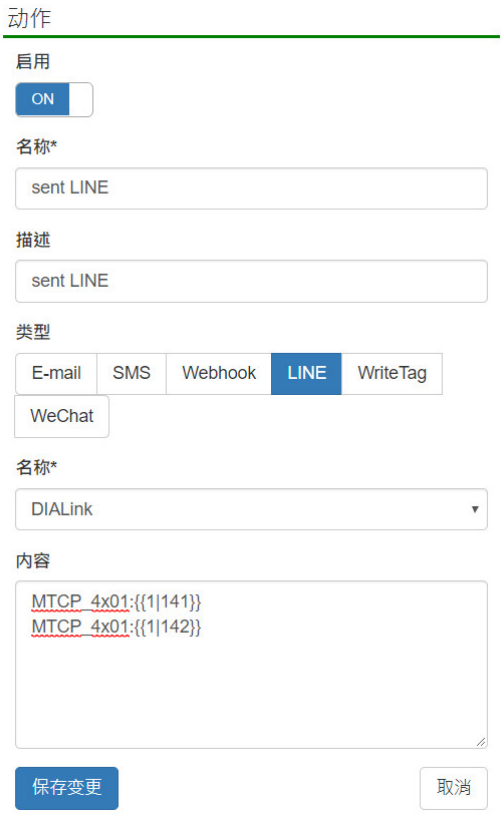
- **新增发送 E-mail 动作**



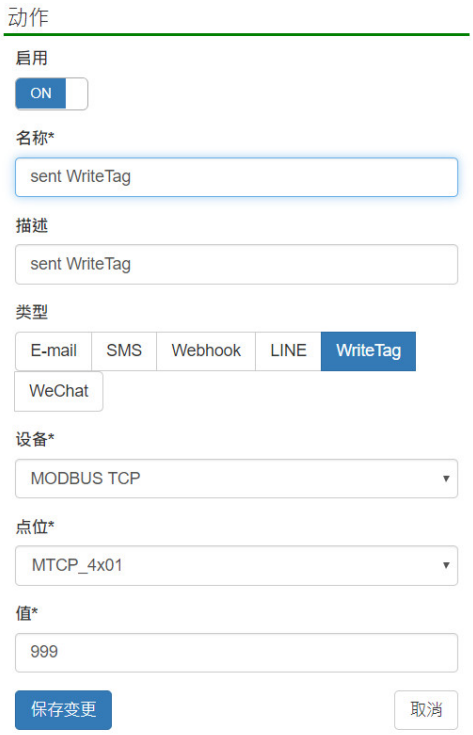
- **新增发送 SMS 动作**



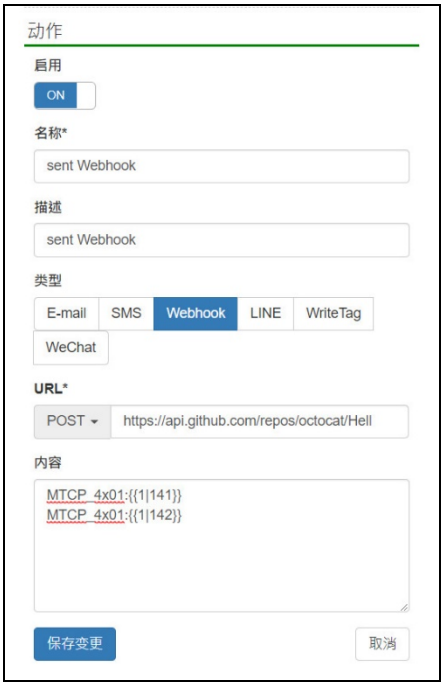
- **新增发送 Line 动作**



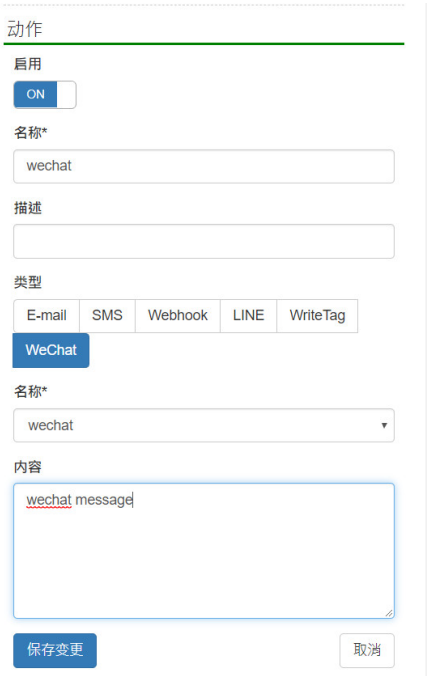
- **新增发送 WriteTag 动作**



- **新增发送 Webhook 动作**



- **新增发送 WeChat 动作**

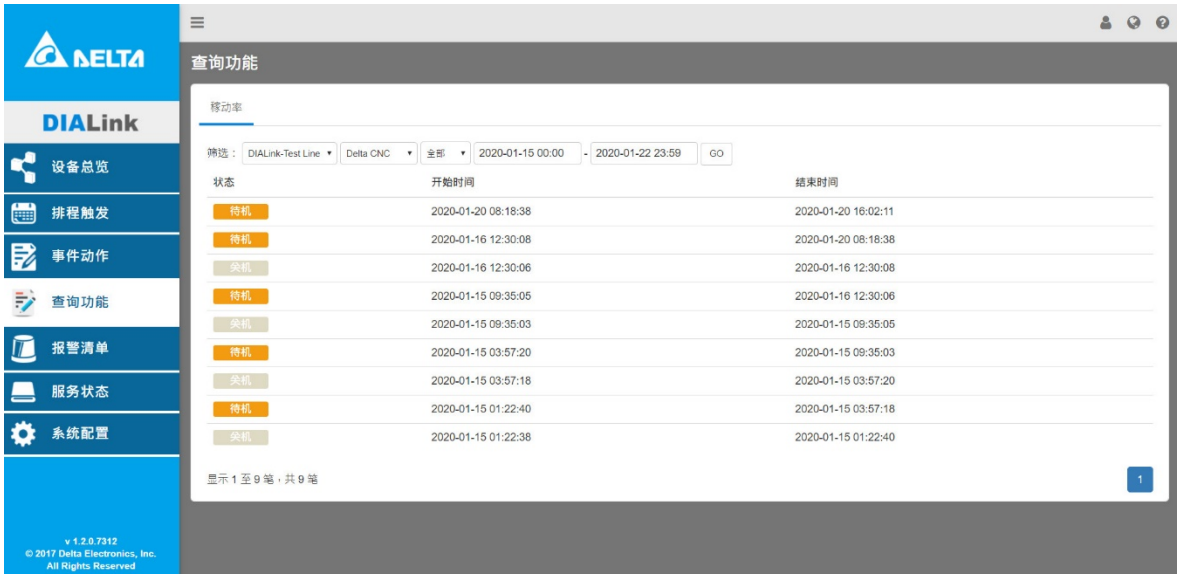


- **新增事件与多笔动作**



2.5 查询功能

【查询功能】页面会列出设备状态维持时间，当状态有改变会再新增一笔纪录。



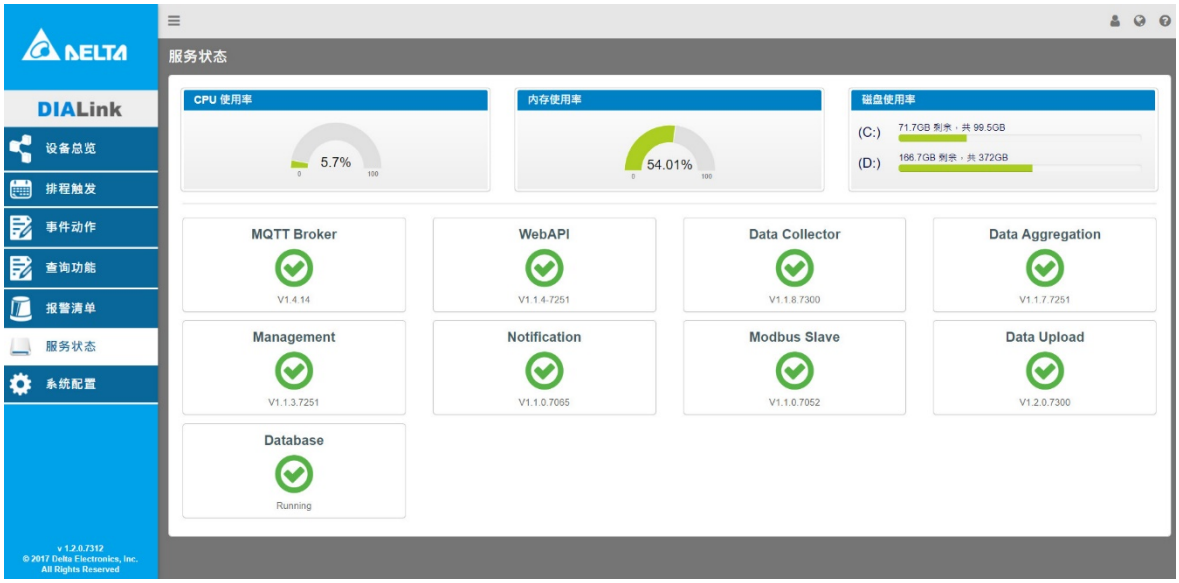
2.6 警报清单

警报列表功能是将所有保存在 DIALink 系统上的事件通过报表方式呈现给用户，可分为节点、设备、点位与系统四大类，用户可根据时间与等级筛选相关警报信息，每页最多显示十笔。



2.7 服务状态

此页面显示主机的 CPU、内存以及磁盘使用率，同时显示 DIALink 系统服务的启动状态，可供用户判断软件哪一服务未启动，系统正常启动时，所有服务会显示绿色勾，状态图标下方显示版本编号；服务未启动时，则会显示红色叉，状态图标下方显示 Not Started；服务尚在启动中时，则会显示灰色循环图标，状态图标下方显示 Checking。若服务全显示未启动，建议进入【Windows】>【任务管理器】>【服务】页面，重新启动【DIALink Service】服务。



以上图标范例中，若服务状态 Data Collector 显示【Not Started】，通常系统判读未合法授权，可尝试检查加密锁是否正确插入主机；Database 则检查 SQL Server 的服务是否正常执行。

2.8 系统配置

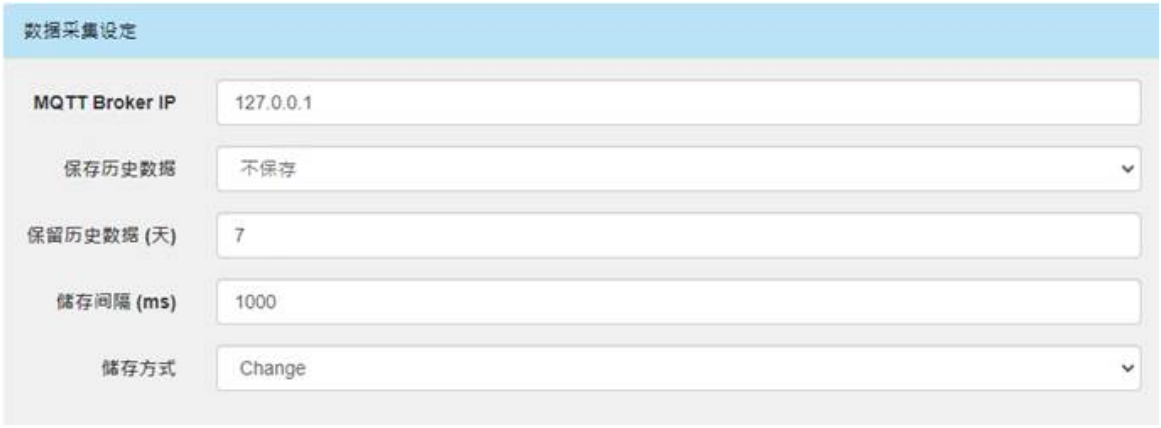
系统配置分八个页面，分别为一般设定、通知设定、网络设定、排班管理设定、LINE 设定、WeChat 设定、E-mail 设定以及 Modbus Slave 设定，用户可根据自身需求设置相关参数。



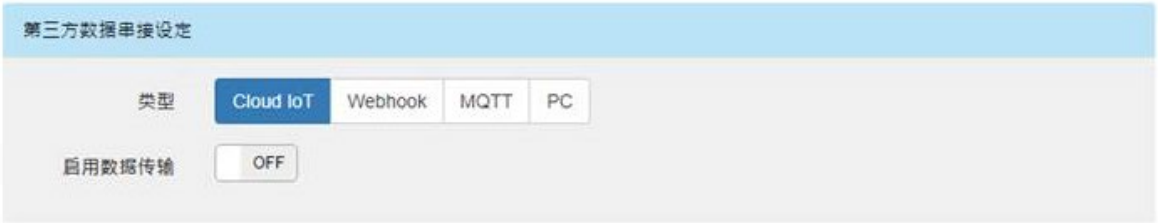
2.8.1 一般设定

一般页面内，可启动 / 关闭数据收集功能、设置 MQTT Broker IP、历史数据保存间隔与存档方式，变更完毕单击【保存变更】。

※历史数据默认为不保存，用户若启动历史数据存档，务必确认硬盘空间是否足够。保存历史数据启动可选择【每日分数据库】或【每月分数据库】或【保存至单一数据库】，此三种选项决定 DIALink 如何在 SQL Server 中创建数据库，对于有需要自行开发软件且有分月/日保存的需求可选择前两种，否则选择【保存至单一数据库】即可。点位的数据可以依照客户需求设置保存数据的方式：Change 是点位数值有变更才会保存到数据库，Interval 则是依照保存间隔时间记录到数据库，保留历史数据（天）则是决定历史数据在几天以前的信息会被删除。但是保存历史数据的机制必须在启动【保存历史数据】才有作用。



于 DIALink 网页上的选单「服务状态」上，可以看到一个模块「Data Upload」，这个模块负责将设备收集上来的数据传到第三方外部系统。此服务是否要运行，可通过【系统配置】、【第三方数据串接设定】来设置开启 / 关闭第三方数据串接功能，目前提供方式有【Cloud IoT】、【Webhook】、【MQTT】及【PC】。由于这部分内容繁多，我们统一在附录 I 进行说明。

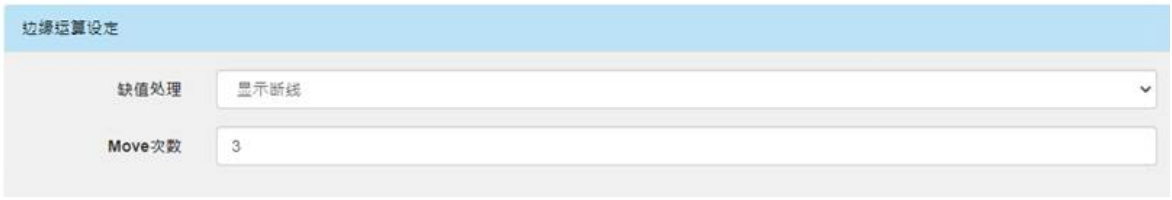


边缘运算设置区块：

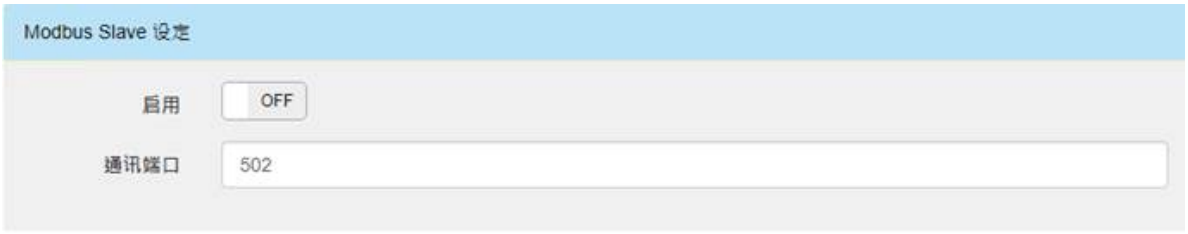
【缺值处理】运算公式中点位读取失败可设置判定条件选项为：【显示断线】及【使用前值】。（详细概念请参考附录 D）

【Move 次数】若在虚拟点位中设置了 Move 功能，在数值产生变化时才会进行 Move，数值没有变化则不会进行 Move 的动作，而 Move 次数则是在数值改变后，持续几次 Move 来保证数值能正确搬移，

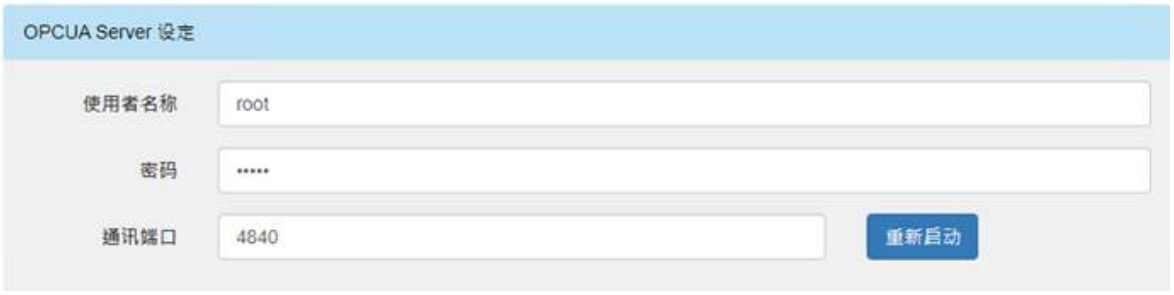
若设置 3 次，则 Move 进行 3 次之后判断数值没有变化就不会再 Move，直到数值又有变化才会再次触发。



开启 / 关闭 Modbus Slave 功能及通讯端口，变更完毕点选【保存变更】。



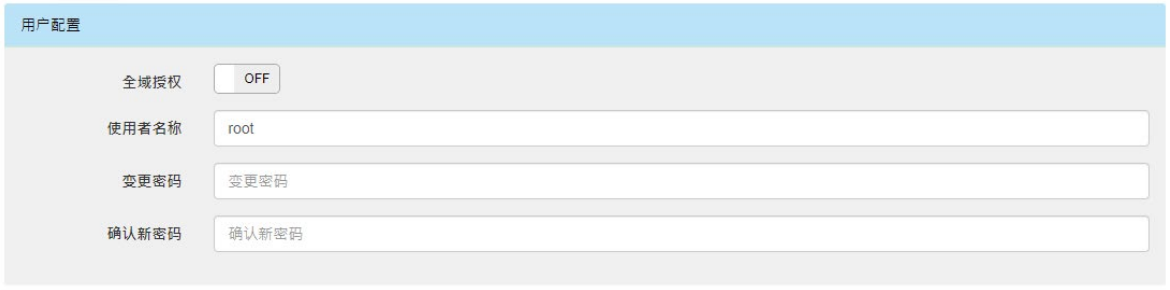
OPCUA Server 提供登入名称、密码及通讯端口的变更。系统默认用户名称 / 密码：root / admin，通讯端口：4840。详细说明请参考附录 E。



使用者设置区块:

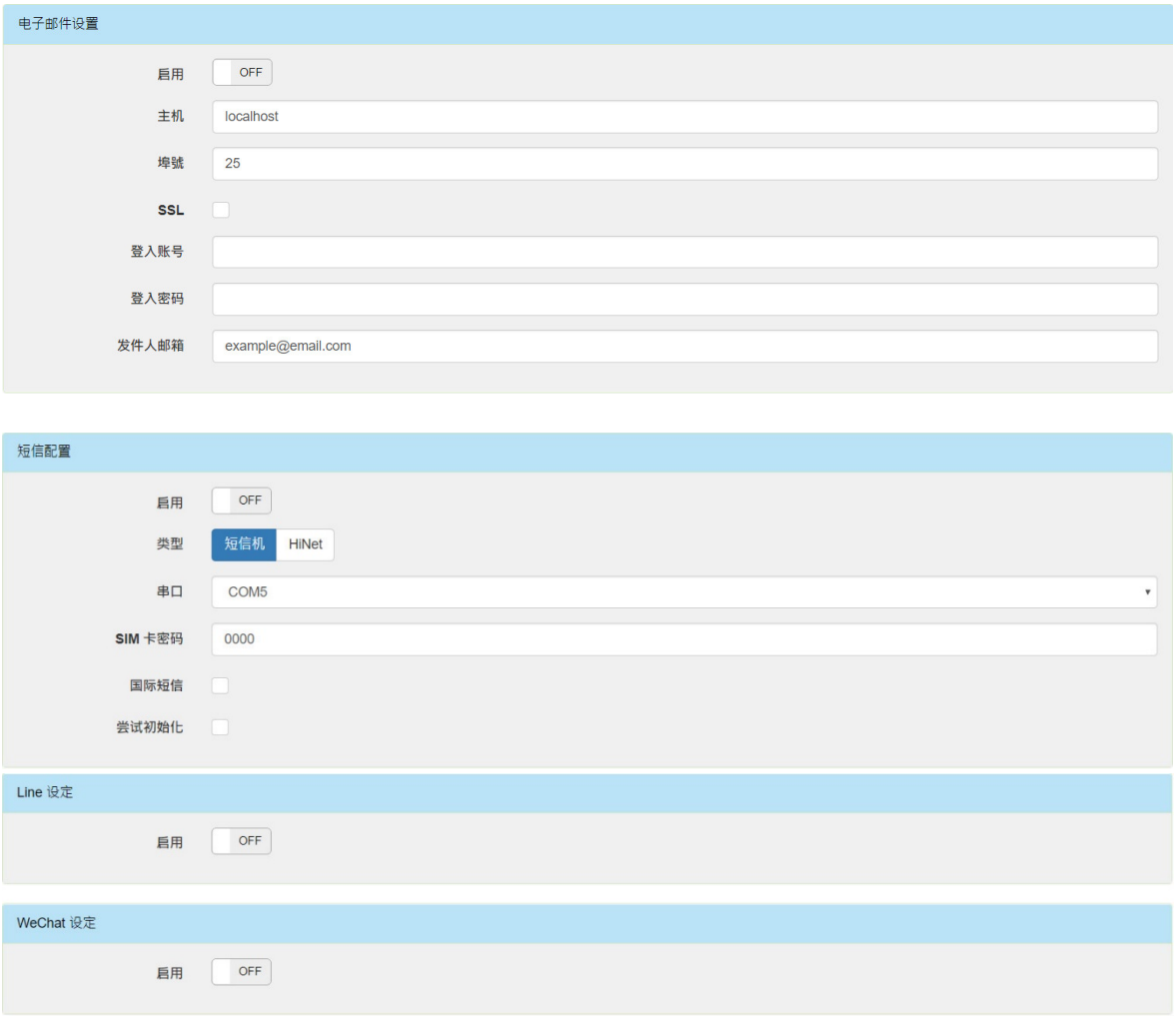
【可启用或关闭全局授权】。请参考附件 H 内容。

【可变更使用者账号密码】，登入帐户限定为一组，请务必牢记使用者账号密码，变更完毕点选【保存变更】。



2.8.2 通知设定

通知页面内，可设置电子邮件的主机参数、短信发送参数、Line 及 WeChat 开关设置，相关数据设置正确，触发点位、排程触发以及事件动作的通知功能方可启用，短信设置又分成外挂的短信机参数及 HiNet 主机参数两款，变更完毕单击【保存变更】。



2.8.3 网络设定

网络页面内，可设置主机 LAN1 & LAN2 网卡 IP 设置，网络接口可选择目前与主机连接的网络，显示网络名称 [MAC Address]，具 IPC 主机的机种，默认 IP 为 192.168.1.1 与 192.168.2.1，变更完毕单击【保存变更】。



2.8.4 排班管理

排班管理页面内❶，可新增 / 编辑多笔排班时段，❷单击上方 ➕ 符号，新增排班信息。编辑排班信息则是至欲编辑列右方❸单击 进行时段信息的编辑。欲删除单一时段信息，可单击动作

字段的 键进行删除。依据排班设置显示目前时段内稼动率于 CNC 信息看板，实时稼动率圆饼图请参考第 2.2.1 节说明。



新增 / 编辑 排班信息

星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。时段可跨天，但只能存在一笔。



2.8.5 Line 设定

Line 设定页面内❶，可新增 / 编辑多笔 Line 通知设置。❷单击上方 ➕ 符号，新增 Line 信息。编辑 Line 信息则是至欲编辑列右方❸单击 ✏ 进行 Line 信息的编辑。欲删除单一 Line 信息，可单击动作字段的 🗑 键进行删除。若该笔 Line 信息于事件中被使用，则不允许删除。



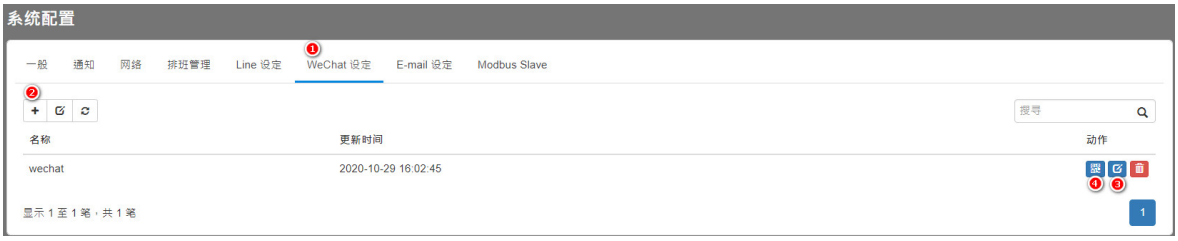
新增 / 编辑 Line 信息

星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。事件名称：必须等同 IFTTT 申请时填入之【Event Name】。Token：IFTTT Webhooks 之密钥，密钥更新必须同步更新此字段内容。IFTTT Line 申请方式请参考附录 B.1。



2.8.6 WeChat 设定

WeChat 设定页面内❶，可新增 / 编辑多笔 WeChat 通知设定。❷单击上方 ➕ 符号，新增 WeChat 信息。编辑 WeChat 信息则是至欲编辑列右方❸单击 ✏ 进行 WeChat 信息的编辑。单击❹可开启 QRCode 加入通知清单。欲删除单一 WeChat 信息，可单击动作字段的 🗑 键进行删除。若该笔 WeChat 信息于事件中被使用，则不允许删除。



新增 / 编辑 WeChat 信息

星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。字段【AppID】、【Secret】、【模版 ID】必须与微信公众号申请时产生之信息相同，系统在保存时会确认是否合法。详细申请方式请参考附录 B.2。



2.8.7 E-mail 设定

E-mail 设定页面内❶，可新增 / 编辑多笔 E-mail 通知设定。❷点选上方 ➕ 符号，新增 E-mail 信息。编辑 E-mail 信息则是至欲编辑列右方❸点选 ✎ 进行 E-mail 信息的编辑。欲删除单一 E-mail 信息，可点选动作字段的 🗑 键进行删除。若该笔 E-mail 信息于事件中被使用，则不允许删除。



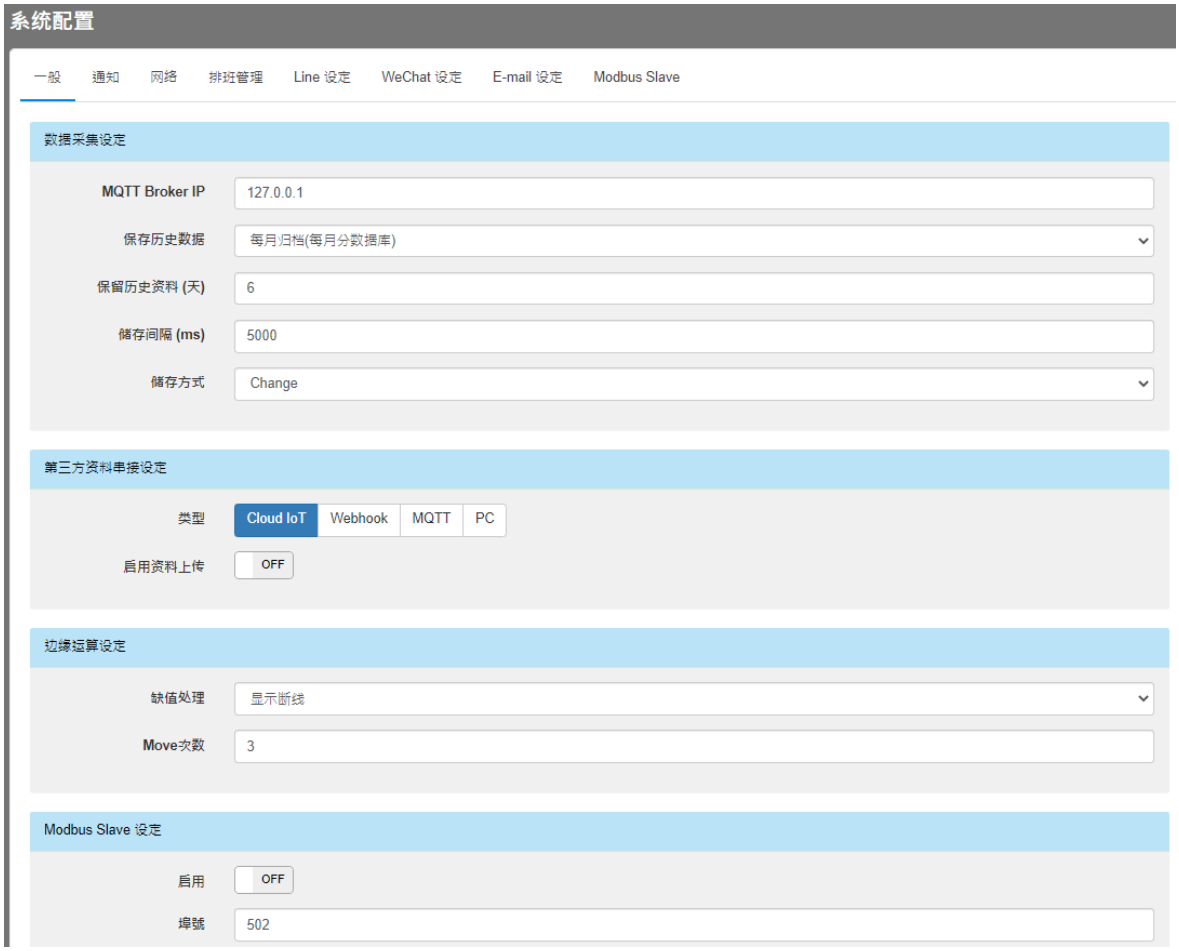
新增 / 编辑 E-mail 信息

星号（ * ）为必填项目，输入完毕点选下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误讯息。

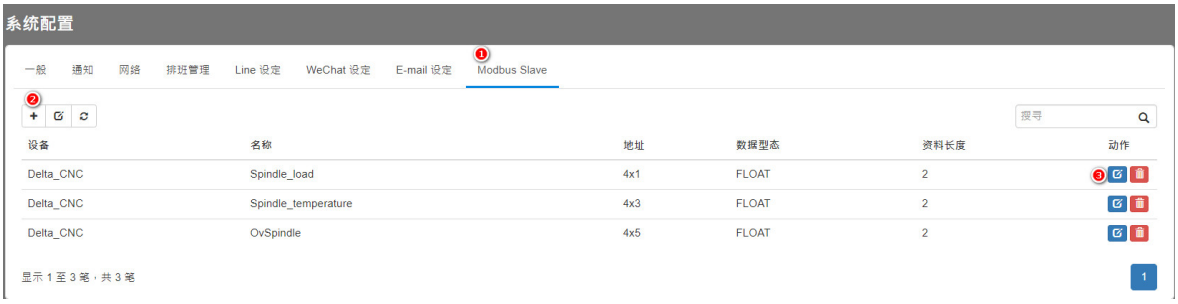


2.8.8 Modbus Slave 设定

DIALink Modbus Slave 联机设定，于系统【一般】页面内开启并设置通讯端口。



Modbus Slave 设定页面内❶，可新增 / 编辑装置点位于寄存器之起始地址及长度。❷单击上方 ➕ 符号，新增点位信息。编辑点位信息则是至欲编辑列右方❸单击 进行点位信息的编辑。欲删除单一点位信息，可单击动作字段的 键进行删除。



新增 / 编辑 Modbus Slave 信息

❶ 单击 [+] 新增一笔或多笔数据。

❷ 新增 Modbus Slave，输入寄存器起始地址（1~65535）及各点位数据长度，点位右方单击 [+]。

❸ 选择建立一笔或多笔数据，按下确定保存变更。

星号（ * ）为必填项目，输入完毕单击下方【保存变更】键，若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。上位程序（MES，SCADA...）存取详细规格说明请参考附录 A。



MEMO

3

第3章 Web API 联机

目录

3.1 概述 3-2

3.2 规格手册 3-3

3.3 范例程序 3-6

3.1 概述

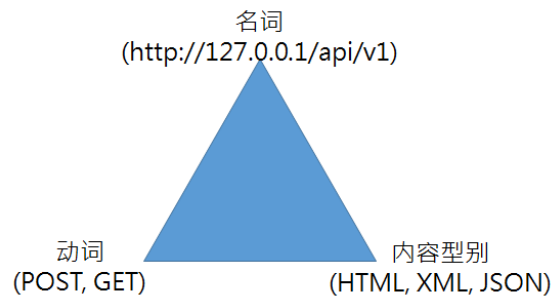
API（Application Programming Interface）是指有特定的定义、传输协议以及进入端口的一个应用程序界面，可供相关人员进行应用程序以及软件的开发。简单来说, API 就是提供一个接口，里面有许多定义好的方法（Method，Function）供程序设计师获取数据。

所谓的 Web API 也就是一个 API 是在 Web 环境下建立，通过 HTTP Protocol 存取数据，Web API 的建立可以使用各种语言，如 JAVA、.NET 等等。目前最常见的 Web API 有 Google APIs，Facebook APIs，用户通过这些 API 来实现各种功能，比如在自己的应用程序或网页画出 Google Map 或 Facebook 登入等。

DIALink Web API 遵循 REST 架构建立 Web API。REST 是一种在因特网服务器与各计算机之间互相交换数据的方法，有别于使用其它如 CORBA、SOAP、WPF 复杂机制的方法，REST 使用简易的 HTTP 传输机制来实现数据交换。若要符合 REST 架构成为一个 RESTful 的 Web API，则需要满足 REST 所定义的五个特性。

- 统一的接口：REST API 的每一个功能称为资源（resource），而每一个资源会有一组独立的地址（URI），通过标准化的 HTTP 接口进行数据交换与通讯。
- 无状态（stateless）：任何一个客户端的交易以及相关信息不会保存在服务器端，也就是不会有 Session 的保存信息。因此，资源之间不会互相影响与关联。
- 可快取：REST API 可定义每一个资源回复给客户端的内容是否可被存放至快取，主要目的在于加速优化效能。
- 主从型态：客户端与服务器端是相互分开的，数据储存是服务器端作业的，客户端无需顾虑数据存放问题，服务器端也无需顾虑客户端的页面呈现方式，因此客户端或服务器端的后续扩充和维护会较为容易。
- 分层化：通过不同阶层的管控，客户端进行联机时无法确定所连接的是中继或是终端计算机，只能确定是否只取得 API 资源。

REST 在资源（resource）的使用上定义了接口（动词 verbs 跟内容类别 content types）、identifiers （名词 nouns）。



从以上三角形图可以了解 REST 的资源（resource）包含了这三个重要项目，在进行任何数据交换之前需先知道当前要使用的资源所需要的三个项目内容。

DIALink Web API 使用了四种类型的动词。第一种方法是＂GET＂，它用于检索或读取资源。第二种方法是＂PUT＂，通常用于更新或替换数据。第三种方法是＂POST＂，用于建立新的资源或记录。第四种方法是＂DELETE＂，这很明显是用来删除记录的。

DIALink 提供 DIALink Web API，用户只需要遵循 API 的相关参数要求就可以得到想要的数据，这是为了符合在大多数情况下用户会需要或者想要自行去取得相关信息产出一份报告或自行分析数据。因此，下面的章节将进一步介绍如何使用 DIALink Web API，并提供一些重要的实作范例。

3.2 规格手册

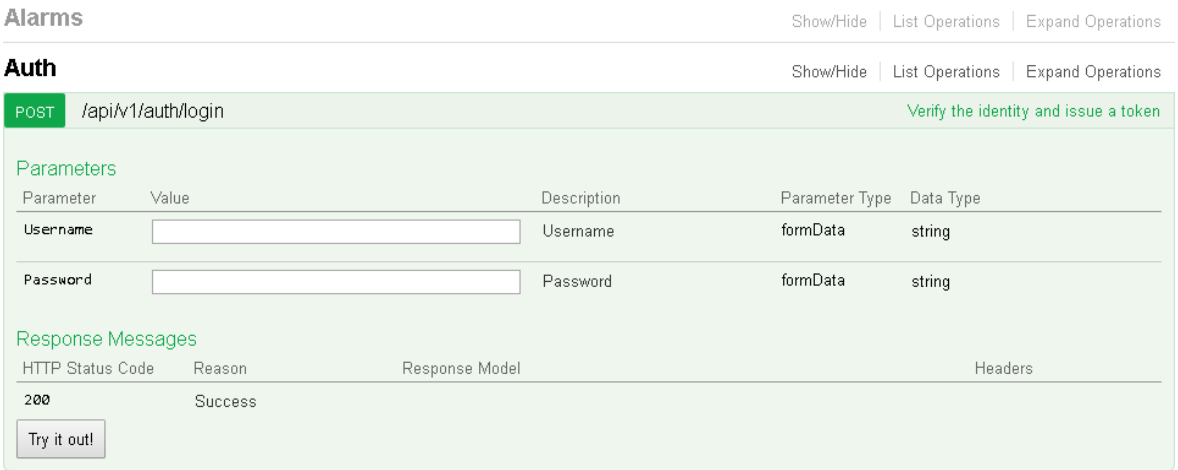
登入 DIALink 后，可在右上方图标下拉菜单上选择 API 文件，如下图所示。



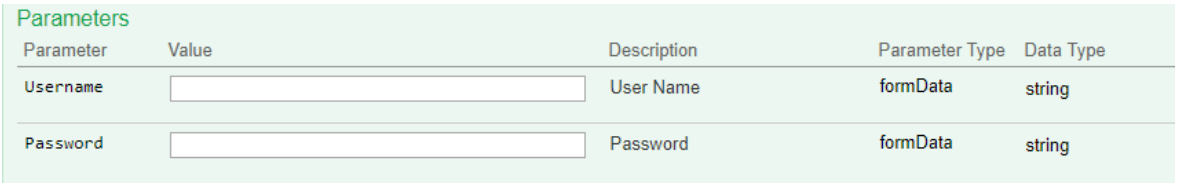
点选 API 文件后，会进到 Swagger UI 的页面



使用鼠标单击页面上的任何一个超链接，例如 Auth / POST，页面会显示更多的信息如下所示。



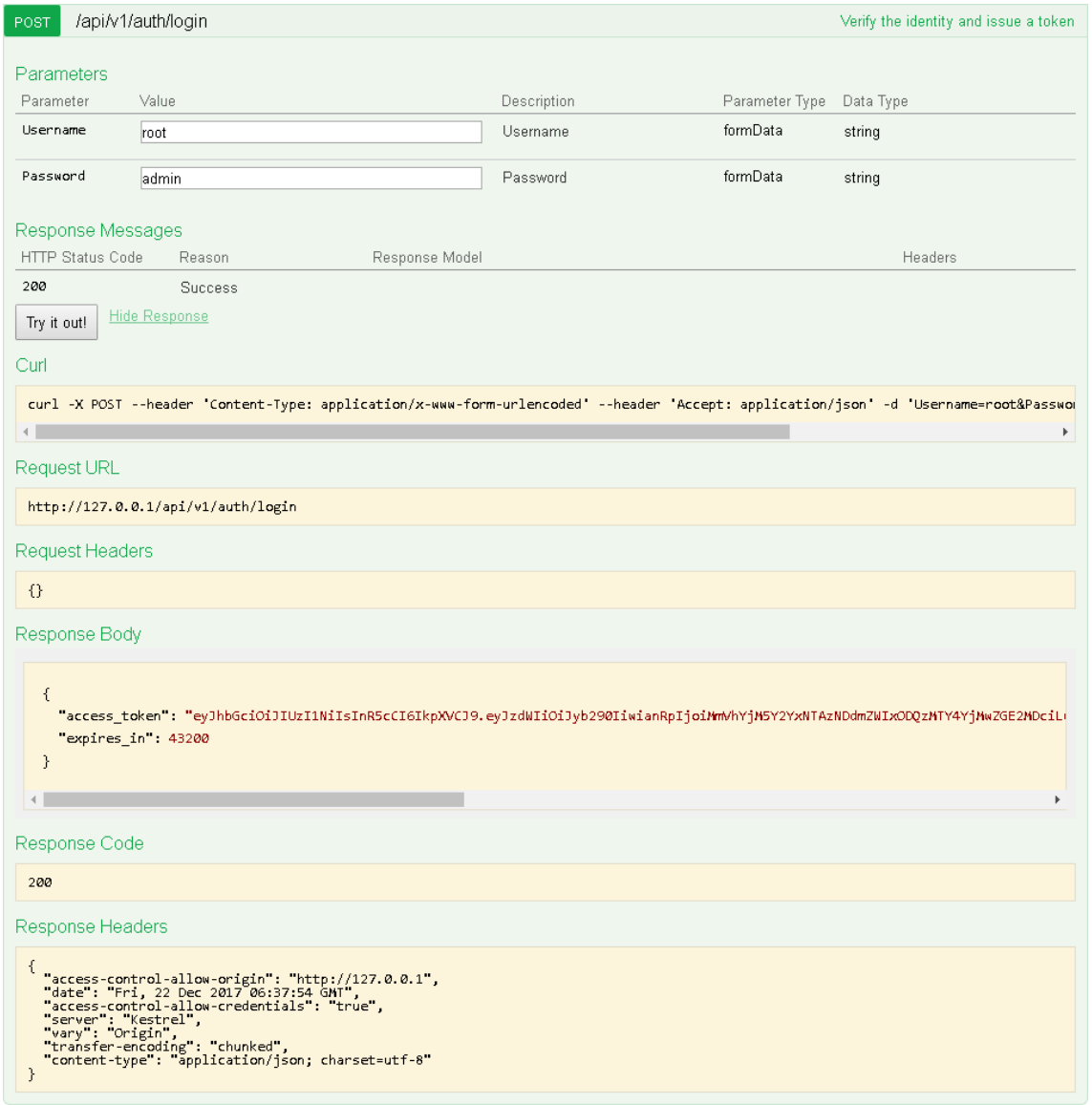
左上角显示的 POST /api/v1/auth/login 是该 API 所使用的 HTTP method，也就是 REST 所表示的动词，接着后方的是名词 URI 网址，完整的 URI 需再加上 IP 或 Domain 才能正确的存取 API。Parameters 的区块显示需要两个参数，参数类型是 formData，数据类型为字符串。



Parameters

Parameter	Value	Description	Parameter Type	Data Type
Username		User Name	formData	string
Password		Password	formData	string

Response Messages 区块显示 HTTP 状态代码和其他信息，Username 与 Password 为 DIALink 登入的账号密码，当点击按钮 " Try it out! " 之后，它显示响应信息如下。



POST /api/v1/auth/login — Verify the identity and issue a token

Parameters

Parameter	Value	Description	Parameter Type	Data Type
Username	root	Username	formData	string
Password	admin	Password	formData	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	Success		

Try it out! Hide Response

Curl

```
curl -X POST --header 'Content-Type: application/x-www-form-urlencoded' --header 'Accept: application/json' -d 'Username=root&Passwo
```

Request URL

```
http://127.0.0.1/api/v1/auth/login
```

Request Headers

```
{}
```

Response Body

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJyb290IiwianRpIjoiMmVhYjM5Y2YxNTAzNDdmZWIxODQzMTY4YjMwZGE2MDciL
  "expires_in": 43200
}
```

Response Code

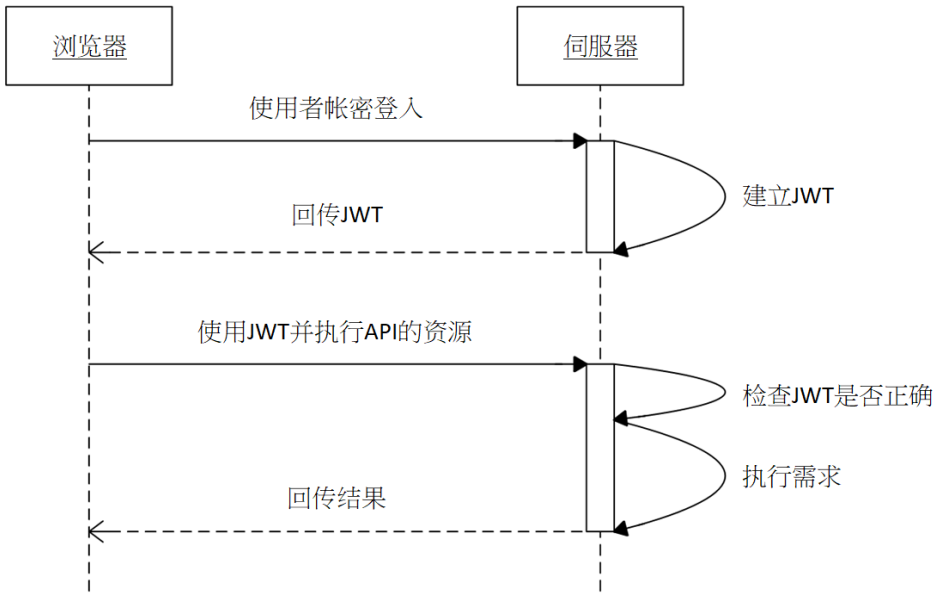
```
200
```

Response Headers

```
{
  "access-control-allow-origin": "http://127.0.0.1",
  "date": "Fri, 22 Dec 2017 06:37:54 GMT",
  "access-control-allow-credentials": "true",
  "server": "Kestrel",
  "vary": "Origin",
  "transfer-encoding": "chunked",
  "content-type": "application/json; charset=utf-8"
}
```

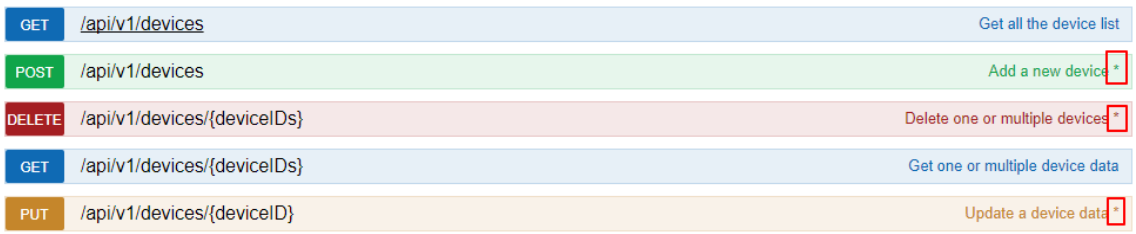
DIALink Web API 所有资源在 Response Code 是 2xx 的时候，表示执行成功；如果有回传的数据内容，则可以在 Response Body 区块看到，类别预设是以 JSON 格式传递。

由于 RESTful 有着无状态（stateless）的特性存在，因此会面临验证跟授权问题。目前已有许多方法可以用来处理这个问题，而 JWT 则是一个相对简单但又安全的方式。什么是 JWT?JWT 全名是 JSON Web Token，使用 JSON 对象并且通过哈希签名来确保数据没有被变更过。因此可以把它当作验证的 token 来用，也可以用来在两个 client 之间传递数据。JWT 成功验证循序图如下。



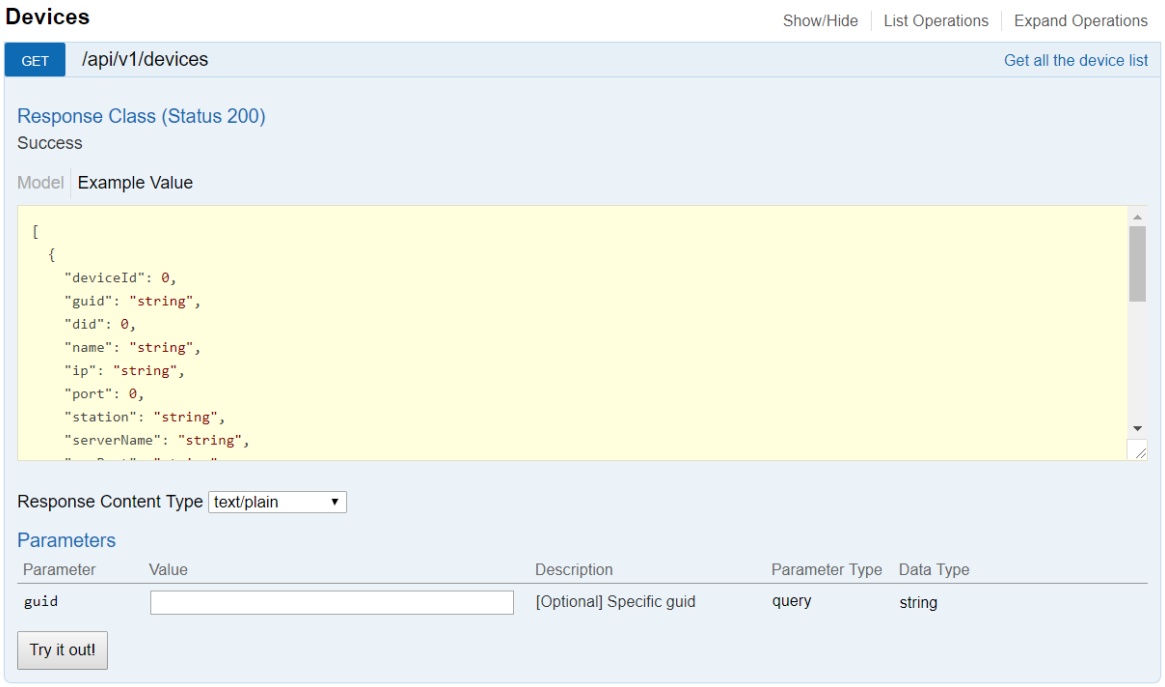
首先使用账号密码进行登入，之后服务器会产生一组 JWT 对象回传给浏览器。在取得 JWT 之后，再进一步执行 API 资源，同时将 JWT 放入到 HTTP HEADER 让服务器能进行 JWT 验证，如果验证失败会被拒绝存取（Status Code：401）。

若用户在线 Swagger 说明网页中看到 API 说明后面有*号（如下图），表示这个 API 需要进行验证，请参考第 3.3 节范例，通过 auth / login 资源取得 token 后，存取时带入 HTTP header 的 " Authorization " 字段中。



3.3 范例程序

本节将介绍实际如何使用 DIALink Web API 资源，并以 C# 程序为范例，首先介绍的是最常使用的 GET 取值范例。

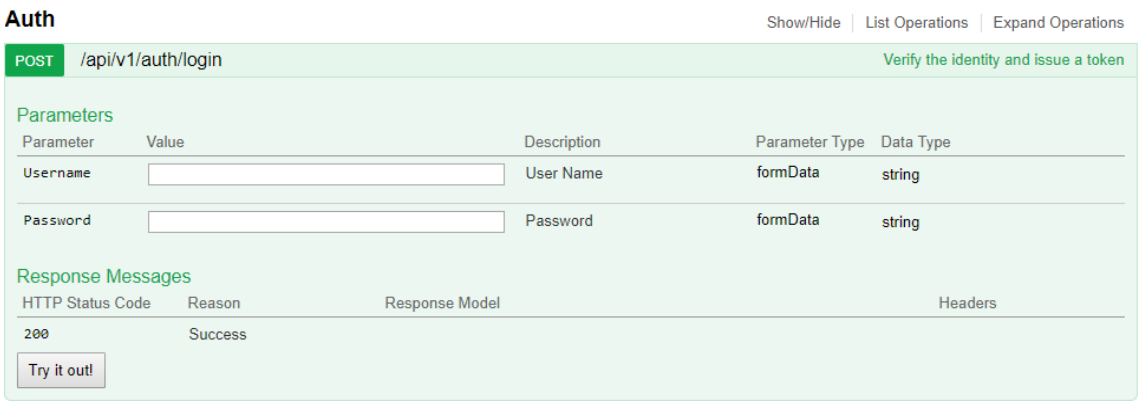


从页面中可以看到这个 API 资源使用 GET 方法、URI 是 / api / v1 / devices、成功代码 200，回传的内容格式是 JSON 类型的内容，请先通过 NuGet 安装使用人数最多的 JSON 格式处理套件 " Newtonsoft.Json " ，如何取得完整内容的程序代码，如下所示 (.NET Framework 4.5 以上) :

```
string sURL = "http://127.0.0.1/api/v1/devices";
using (var client = new HttpClient())
{
    HttpResponseMessage response = client.GetAsync(sURL).Result; // 使用 GET 方法
    if (response.IsSuccessStatusCode) // 判断成功状态代码 2xx
    {
        string strjson = response.Content.ReadAsStringAsync().Result; // 读取响应内容
        dynamic items = JsonConvert.DeserializeObject(strjson); // 将 JSON 转为对象
        foreach (var item in items)
        {
            string guid = item["guid"]; // 取得 guid
            // 取得其它字段
        }
    }
}
```

DIALink Web API 目前大部分的 POST、PUT 和 DELETE 都需要先取得授权状态后才能使用。但若在【系统设定】中开启了选项：启用全域授权（请参考附录 H）后，则所有的 Web API 都要先取得授权后才能使用。

因此，了解如何获得授权是关键，首先看到 Web API 的 Login 资源页面。



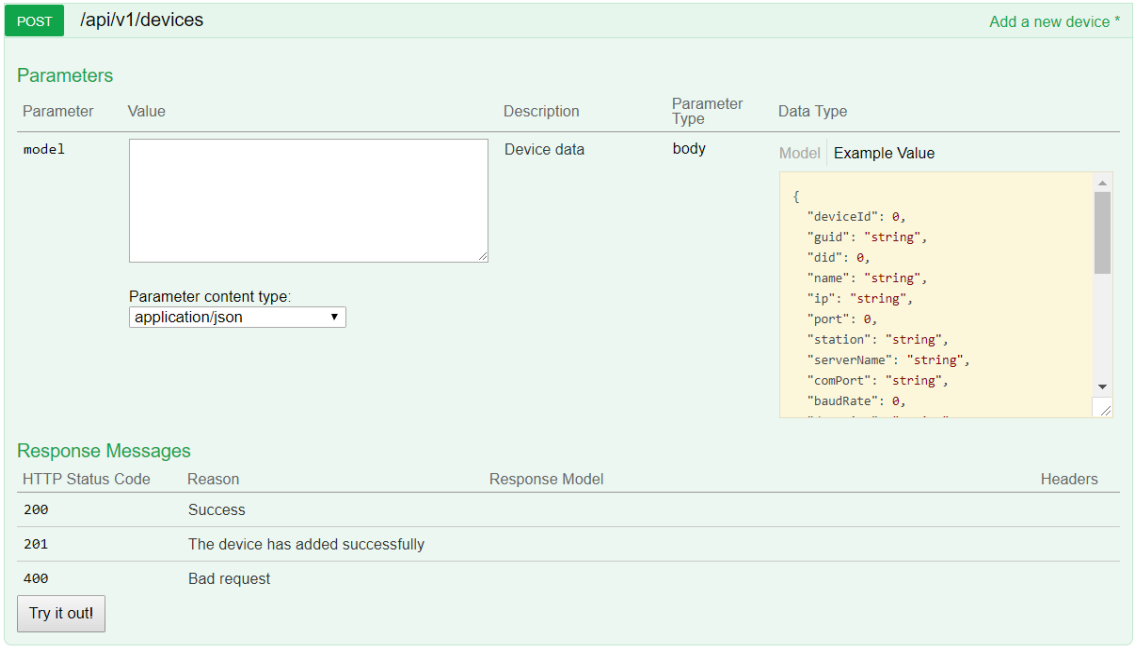
从页面中得知这个 API 的资源使用到 POST 的方法、URI、以及需要带入两个字符串参数而成功的响应代码是 200，有了以上信息就可以通过程序代码来取得授权码，程序代码如下。

```
string authorizationToke = "";
    string sURL = "http://127.0.0.1/api/v1/auth/login";
    using (var client = new HttpClient())
    {
        var formContent = new FormUrlEncodedContent(new[] { new KeyValuePair<string, string>("Username", "root"), new KeyValuePair<string, string>("Password", "admin") }); // 账号密码
        HttpResponseMessage response = client.PostAsync(sURL, formContent).Result; // 使用 POST 方法
        if (response.IsSuccessStatusCode)
        {
            string strjson = response.Content.ReadAsStringAsync().Result;
            dynamic item = JsonConvert.DeserializeObject(strjson);
            authorizationToke = item["access_token"]; // 取得 access_token
            int expired = Convert.ToInt32(item["expires_in"]); // 有效期限
        }
    }
```

程序代码内的 item 可以获得 access_token，这是 Web API 所需要的授权码（token）。在每一次取得 access_token 的时候都会一并取得一个期限时间（expires_in，单位秒），超过时间，这组 token 就会自动失效，因此如果用户程序需要持续存取 DIALink 数据，请在 token 过期之前重新获取新的 access_token。取得 access_token 之后，下一个范例会说明如何将 access_token 放入到 HTTP

header。

下图是需要用户许可证的 API 资源，URI 是 " / api / v1 / devices "（右上角说明后面有 * 号表示此 API 需要验证）。



程序代码如下：

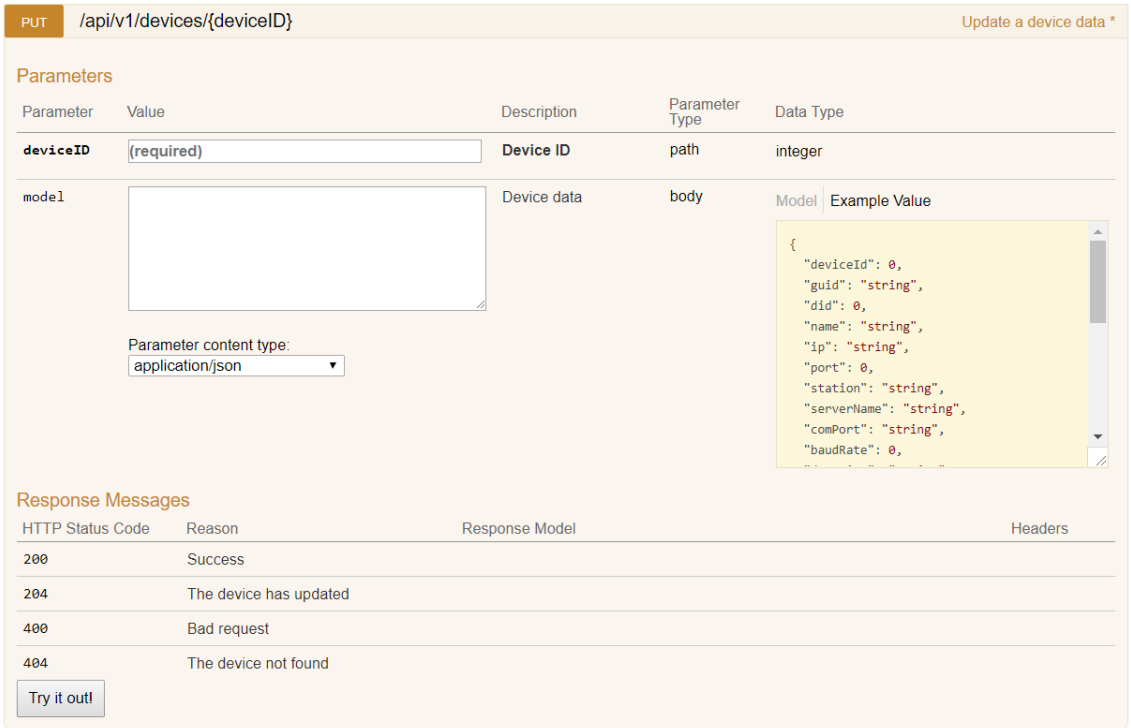
```
string sURL = "http://127.0.0.1/api/v1/devices";
    var device = new // 建立数据对象，根据实际情况选择填入的字段
    {
        deviceId = 0, // 新增时 deviceId 会自动新增，因此带入 0(或不设定)即可
        guid = "968644f3a1294c2f9da03b4af1a473ac", // 新增到某一台 DIALink 的唯一标识符
        name = "Modbus_Test2",
        ip = "127.0.0.1",
        port = 502,
        station = "1",
        commType = 0, // TCP 联机类型
        type = "Modbus",
        brand = "Modbus",
        model = "Modebus",
        connectEnabled = true,
        comment = "API 建立测试"
    };

    var strjson = JsonConvert.SerializeObject(device); // 将对象转成 JSON
    using (var client = new HttpClient())
```

```
{
    client.DefaultRequestHeaders.Add("Authorization", "Bearer " + authorizationToken);
    var content = new StringContent(strjson, Encoding.UTF8, "application/json");
    HttpResponseMessage response = client.PostAsync(sURL, content).Result; // 使用
POST 方法
    if (response.IsSuccessStatusCode)
    {
        Console.WriteLine("create success");
    }
    else
    {
        Console.WriteLine(response.ReasonPhrase);
    }
}
```

guid 代表一台 DIALink 的唯一标识符，在这台 DIALink 下能够增加许多设备，这个范例中是增加一台新设备。首先需要注意的是 HTTP authorization header，它是使用 bearer token 来获得许可，而 authorizationToken 是在前一个范例中取得的值。使用任何一种 API 资源都必须注意是否需要授权，特别是使用到 POST、PUT 以及 DELETE 的资源。其次，内容类型必须遵循 API 的要求。最后，参数是 JSON 格式的数据，而 JSON 的内容可以在 Web API 页面的 Parameters 的区块找到。

接着尝试修改刚刚新增的数据，假设刚刚建立起来的 deviceID 代码为 5，可以通过 PUT 更新设备内容。



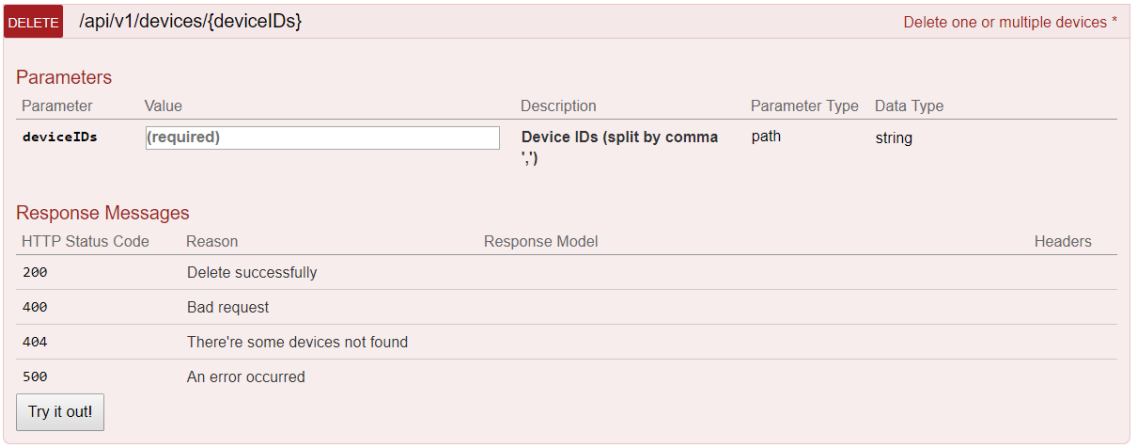
从图上可以看到这个 API 资源多了刚刚 POST 所没有的 deviceID 参数，假设刚新增的 deviceID 是 5，则把大括号的{deviceID}取代为数字，而要传送的 model 内容依然沿用 POST 的方式加以修改即可，程序代码如下：

```
string sURL = "http://127.0.0.1/api/v1/devices/{deviceID}";
    sURL = sURL.Replace("{deviceID}", "5");
    var device = new
    {
        guid = "968644f3a1294c2f9da03b4af1a473ac",
        name = "Modbus_Test3", // 修改名称
        commType = 1, // 修改联机类型为 Serial Port
        comPort = 4, // COM4
        baudRate = 115200,
        dataBits = 8,
        parity = "N",
        stopBits = 1,
        mode = 0,
        type = "Modbus",
        brand = "Modbus",
        model = "Modebus",
        connectEnabled = true,
        comment = "API 修改测试"
    };

    var strjson = JsonConvert.SerializeObject(device);
    var content = new StringContent(strjson, Encoding.UTF8, "application/json");
    using (var client = new HttpClient())
    {
        client.DefaultRequestHeaders.Add("Authorization", "Bearer " + authorizationToken);
        HttpResponseMessage response = client.PutAsync(sURL, content).Result; // 使用 PUT
方法
        if (response.IsSuccessStatusCode)
        {
            Console.WriteLine("modify success");
        }
        else
        {
            Console.WriteLine(response.ReasonPhrase);
        }
```

}

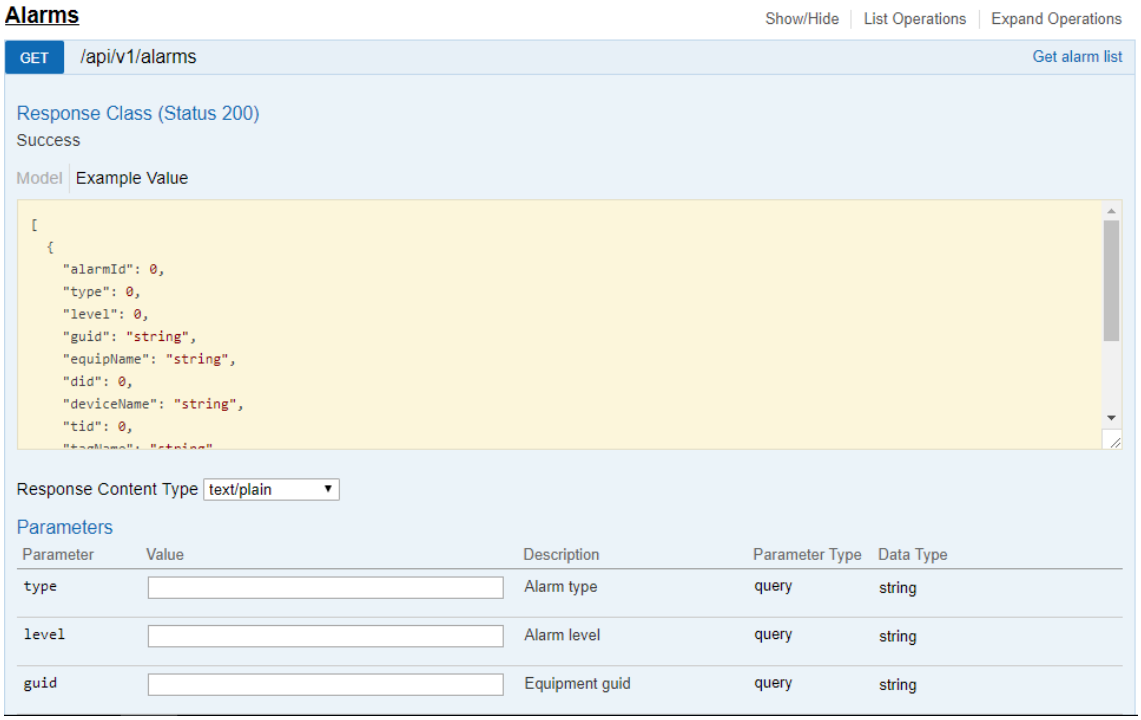
接着尝试把刚加入的 device 删除，请看下图 Web API 的删除信息的详细内容。



从这个 API 资源中可看到同样需要 deviceID 参数，这样只需要将大括号取代后传送出去就可以删除信息，范例程序代码如下。

```
string sURL = "http://127.0.0.1/api/v1/devices/{deviceID}";
    sURL = sURL.Replace("{deviceID}", "5");
    using (var client = new HttpClient())
    {
        client.DefaultRequestHeaders.Add("Authorization", "Bearer " + authorizationToken);
        HttpResponseMessage response = client.DeleteAsync(sURL).Result; // 使用 Delete 方法
        if (response.IsSuccessStatusCode)
        {
            Console.WriteLine("delete success");
        }
        else
        {
            Console.WriteLine(response.ReasonPhrase);
        }
    }
```

最后来看在 DIALink Web API，GET 的资源参数种类是 query。



与 path 不同的地方是 query 在 URI 后方加上问号以及参数名称和参数值为选择性的参数，下面使用 level、startTime 以及 endTime 取得特定时间范围的警报来作为程序代码范例的参数。

```
string sURL = "http://127.0.0.1/api/v1/alarms?level=all&startTime=2017-12-22
00:00:00&endTime=2017-12-23 23:59:59";
    using (var client = new HttpClient())
    {
        HttpResponseMessage response = client.GetAsync(sURL).Result; // 使用 GET 方法
        if (response.IsSuccessStatusCode)
        {
            string strjson = response.Content.ReadAsStringAsync().Result;
            dynamic items = JsonConvert.DeserializeObject(strjson);
            foreach (var item in items)
            {
                string message = item["message"]; // 取得警报信息
            }
        }
    }
```

4

第4章 MQTT 联机

目录

4.1 概述 4-2
4.2 规格手册 4-2
4.3 程序范例 4-5

4.1 概述

MQ Telemetry Transport（MQTT）是 IBM 信息传递技术中的一种，主要目的是为了解决低带宽（Low-Bandwidth）以及高网络等待时间（High-Latency）。设计原则是尽量减少网络带宽和设备资源需求，同时要确保可靠性和一定程度的交付保证。这些原则也使得协议成为连接设备“机器对机器”（M2M）或“物联网”以及带宽和资源有限设备的理想选择。目前有许多竞争的 IIoT 技术和协议在起作用，而 MQTT 极轻量级开销（2-byte header），发布（Publish）/ 订阅（Subscribe）模型和双向（bi-directional）能力可支持小型测量仪或监控器，以及在远距离、不稳定的网络环境传输数据，这是它满足工业控制系统需求的独特之处。

在 2013 年，IBM 向 OASIS 技术委员会提交了 MQTT，使得 MQTT 成为一个开放协议和标准。由于 MQTT 成为开放协议与标准，所以很容易被应用于各种物联网设备。许多应用程序也可以通过使用相对应的 MQTT Client 扩充套件来实现 MQTT 通讯。

4.2 规格手册

在使用 MQTT 之前需要了解几个 MQTT 所提到的名词。

QoS：服务质量（Quality of Service）级别明确规范了每个 MQTT 信息的传送方式，信息传递有三种 QoS 可以使用。

QoS 0（最多一次）- 根据操作环境努力传递信息。信息有可能丢失。

QoS 1（至少一次）- 确保信息到达，但有可能发生重复。

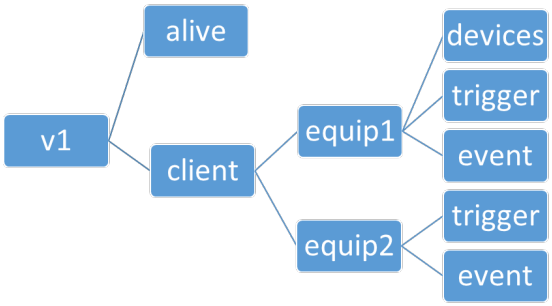
QoS 2（恰好一次）- 确保信息到达一次。

QoS 的特性有一个简单的规则就是服务质量越高效能越低，每个 MQTT 信息传送要依据环境以及功能选择最适合的 QoS 才能达到最高效率与稳定性。

Broker：简单来说 Broker 是 MQTT 的伺服端，它是处理任何发布 / 订阅协议的核心。Broker 可以处理多达数千个并发连接的 MQTT 客户端，负责接收所有信息以及过滤，然后将信息发送给所有订阅的客户端。它也持有所有客户端的信息，包括订阅和错过的信息。而 Broker 另一个责任是客户的认证和授权。

Client：客户端（Client）可以是任何计算机、任何一个小型的嵌入式系统或资源受限的设备，基本上任何具有TCP / IP通讯能力以及运行MQTT的通讯协议并透过任何类型的网络连接到MQTT Broker，这就是为什么MQTT非常适合小型设备。客户端要实现MQTT的通讯协议非常简单，目前MQTT Client 类别库可用于各种编程语言，例如Android、C、C ++、C # 、iOS、Java、JavaScript以及.NET等。MQTT客户端可以同时是发布者(Publisher)和订阅者(Subscriber)。

Topics：在发布（Publish）/ 订阅（Subscribe）信息，信息内容目的地被称为主题（Topic），客户端可以发布主题也可以订阅主题。而在 MQTT 的协议中 Topics 需具备阶层的概念，这表示了 Topics 要依据这样的结构来建立，这样才能确保订阅者和发布者能明确表示主题。主题是自由格式的字符串，可以由任何字符组成。然而，符号「/」、「+」与「#」具有特殊含义，因此不可随意使用，需要特别注意。



由上图可看出这是一个阶层图，而方块内容就是特定的主题。如果今天想订阅 alive 这个主题，那么完整的主题会是 “v1 / alive”，由此可知符号 / 是用来区分阶层。前面还有提到的 + 与 # 分别表示单一阶层与多重阶层的意思，首先来看何为单一阶层，如果今天订阅的主题是 v1 / + 那么符合这项的只有 v1 / alive 与 v1 / client，而 v1 / client 之后的阶层都不成立。

☑v1/alive
☑v1/client
☒v1/client/equip1
☒v1/client/equip2/device

多重阶层指的是假如今天订阅主题为 v1 / # 那么所有在 v1 之下的主题都会收到。

☑v1/alive
☑v1/client
☑v1/client/equip1
☑v1/client/equpi2/device

资安凭证：

在用户选择了 CA 模式来安装 DIALink 或者需要手动变更网页的安全层级时，请参考附录 H，在使用凭证的设定下，需确认安装路径（默认为 C:\DIALink）下 config 文件夹内的 mosquitto.conf 需要设定用户凭证文档路径，并确认该路径下确实有相关凭证文档。

```
# wss: Web Socket with TLS
# wss listener port-number [ip address/host name]
# CA certificate
cafile C:/DIALink/Web/CA/mqtt/DIALink_CA.crt

# Path to the PEM encoded server certificate.
certfile C:/DIALink/Web/CA/mqtt/DIALink.crt

# Path to the PEM encoded keyfile.
keyfile C:/DIALink/Web/CA/mqtt/DIALink.key
```

通过 MQTT 进行数据传输，需加入认证联机信息，系统开放用户通过 MQTT 链接的用户名 / 密码，分别为 **user / dialuser**，可参考下方使用语法与第 4.3 节程序范例：

```
mqttClient.Connect("client_id", "user", "dialuser", false, 60);
```

下表为 DIALink 在 MQTT 中定义的主题

Topic	QoS	Description	Sample	Note
v1/alive	0	DIALink heartbeat and information	{ "id": "13a17c0cc2f4497fb5d6ff473865bfdf", "equipmentType": "CNC", "tagLicence": 1000, "deviceLicence": 5, "status": 1 }	id: DIALink guid deviceLicence: amount of device licence tagLicence: amount of tag licence status: 0:disconnect, 1:connected
v1/client/{guid} devices	1	All device status	[{ "did": 1, "status": 1, "almcode": 0, "almmsg": "", "ts": "2017-07-04T09:22:15.785" }, { "did": 2, "status": 0, "almcode": 1, "almmsg": "Alarm", "ts": "2017-07-04T09:23:15.001" }]	did: device id status: 0:disconnect, 1:connected almcode: alarm code (if alarm occurred) almmsg: alarm message (if alarm occurred)
v1/client/{guid}/tags/{deviceID}	1	All tag status and value by device	[{ "tid": 1, "status": 1, "result": "17046", "ts": "2017-07-04T09:48:02.552", "record": 1 }, { "tid": 2, "status": 1, "result": "9884", "ts": "2017-07-04T09:48:02.552", "record":0 }]	tid: tag id status: 0:disconnect, 1:connected result: tag value record: when 1, the tag valve will save historical data

Topic	QoS	Description	Sample	Note
v1/client/{guid}/lost	2	DIALink lost (disconnect)		(Using "Last Will and Testament")
v1/notification/rpc/request/email	2	Send an E-mail notification	{ "requestId": "fb68281b-a95c-458e-8788-d006e32879cd", "from": "KK", "sendTo": "KUAN.KW.LEE@DELTAWW.COM", "subject": "Test", "content": "TEST EMAIL" }	requestId: Generate a unique ID for check the response result
v1/notification/rpc/request/sms	2	Send a SMS notification	{ "requestId": "fb68281b-a95c-458e-8788-d007e31873cd", "from": "KK", "sendTo": "0912345678", "content": "TEST SMS" }	requestId: Generate a unique ID for check the response result.
v1/notification/rpc/request/webhook	2	Send a Webhook notification	{ "requestId":"fb68281b-a95c-458e-8788-d000e31073cd", "from":"kuan", "uri":"http://172.16.152.79:5000/api/v1/equipment", "method":"get", "content":"TEST Webhook" }	requestId: Generate a unique ID for check the response result. uri: entry your API method: post/get/etc.. depend on your API
V1/notification/rpc/response	1	Notification response result	{ "requestId":"as317c0cc2f4497fb5d6ff473865bfdf", "status":1, "msg":"result message" }	requestId: Generated ID status: 1:successful, 2:fail msg: result message

备注：Tag 数值有变化才会丢出 MQTT 封包，如果某个 Device 内所有 Tag 数值都没有改变，则不会丢出 MQTT 封包。若您联机成功并完成订阅后无法收到任何信息，可将以下两点先厘清：

1. Topic 订阅文字或 GUID 错误。
2. 所有 Tag 数值都没有改变。

4.3 程序范例

范例中使用的MQTT Client 为 M2Mqtt.Net 所提供的对象，本节的范例需在NuGet搜寻 M2Mqtt 及 Newtonsoft.Json 安装。

连接 MQTT Broker：

```
MqttClient mqttClient = new MqttClient("127.0.0.1"); // MQTT Broker IP

string clientId = Guid.NewGuid().ToString(); // 产生随机 id，避免重复
byte retcode = mqttClient.Connect(clientId, "user", "dialuser", false, 60); // 使用帐户密码联机
if (retcode == MqttMsgConnack.CONN_ACCEPTED)
{
    Console.WriteLine("Connect to MQTT broker success"); // 联机成功
}
else
{
    Console.WriteLine("Connect to MQTT broker fail"); // 联机失败
}
```

注册 Topic (Subscribe)：

```
MqttClient mqttClient = new MqttClient("127.0.0.1");
mqttClient.MqttMsgPublishReceived += Client_MqttMsgPublishReceived; // 接收 topic 信息的方法

string clientId = Guid.NewGuid().ToString();
byte retcode = mqttClient.Connect(clientId, "user", "dialuser", false, 60);
if (retcode == MqttMsgConnack.CONN_ACCEPTED)
{
    // 联机成功，订阅 topic
    mqttClient.Subscribe(
        new string[] { "v1/alive", "v1/client/+/devices" }, // topic list
        new byte[] { 0, 1 } // QoS list
    );
}
else
```

```
{
    Console.WriteLine("Connect to MQTT broker fail"); // 联机失败
}
```

接收信息：

```
private void Client_MqttMsgPublishReceived(object sender, MqttMsgPublishEventArgs e)
{
    Console.WriteLine(e.Topic); // 接收信息的 topic
    string payload = Encoding.UTF8.GetString(e.Message); // 接收信息内容
    // 进一步处理接收到的 JSON 数据(需 Newtonsoft.Json)
    dynamic data = JsonConvert.DeserializeObject(payload);
    // ...
}
```

发布信息（Publish）：

发送 E-mail 通知（需先设定好 E-mail 主机信息）

```
MqttClient mqttClient = new MqttClient("127.0.0.1");

string clientId = Guid.NewGuid().ToString();
byte retcode = mqttClient.Connect(clientId, "user", "dialuser", false, 60);
if (retcode == MqttMsgConnack.CONN_ACCEPTED)
{
    string topic = "v1/notification/rpc/request/email";
    var dataObject = new
    {
        requestId = Guid.NewGuid(),
        sendTo = "test@example.com",
        subject = "TEST MQTT SEND EMAIL",
        content = "TEST 123 456 789"
    };
    string json = JsonConvert.SerializeObject(dataObject); // 将数据对象转为 JSON
    mqttClient.Publish(topic, Encoding.UTF8.GetBytes(json), 2, false); // 发布信息
}
else
{
    Console.WriteLine("Connect to MQTT broker fail"); // 联机失败
}
```

MEMO

5

第5章 故障排除

目录

5.1 软件故障排除 5-2
5.1.1 无法登入 DIALink 5-2
5.1.2 MQTT 服务状态显示 Not Started 5-5
5.1.3 Data Collector 服务状态显示 Not Started 5-7
5.1.4 Data Aggregation 服务状态显示 Not Started 5-11
5.1.5 Management 服务状态显示 Not Started 5-13
5.1.6 Notification 服务状态显示 Not Started 5-16
5.1.7 Database 服务状态显示 Not Started 5-18
5.1.8 数据库依旧无法正常运作 5-20
5.1.9 插上硬件 KEY 发生异常，仅能使用试用版 5-26
5.1.10 无法联机到 CNC 设备 5-28

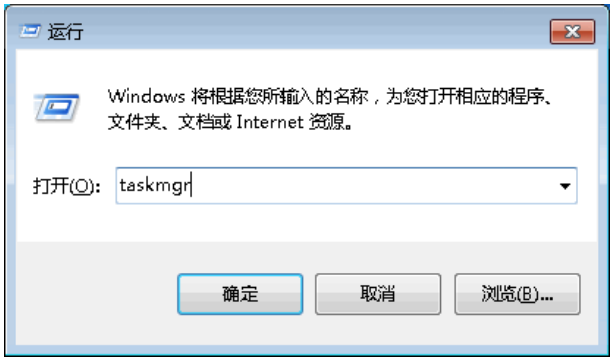
5.1 软件故障排除

5.1.1 无法登入 DIALink

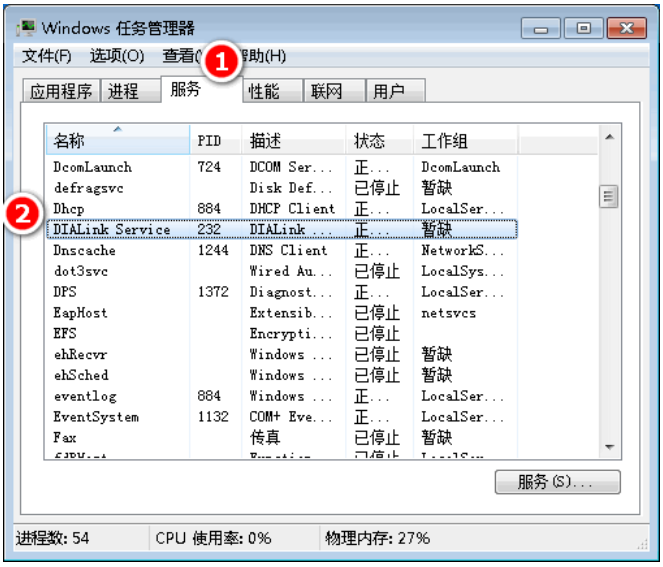


- **处理程序：确认任务管理器内的 DIALink 服务是否启动**

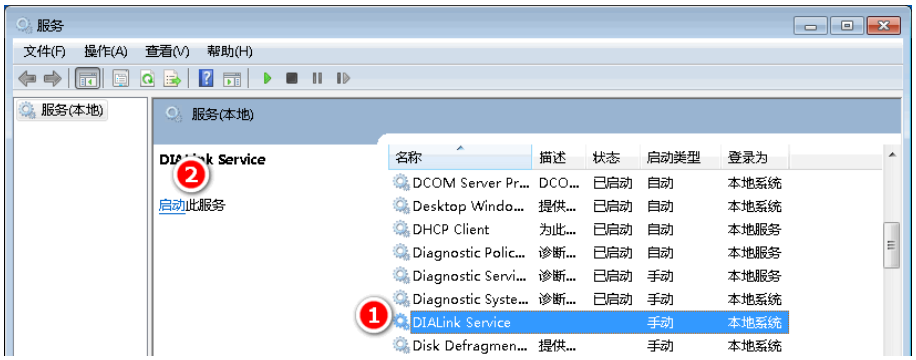
步骤一：按 键 +R 或【开始】> 【执行】，输入 taskmgr 进入 Windows 任务管理器。

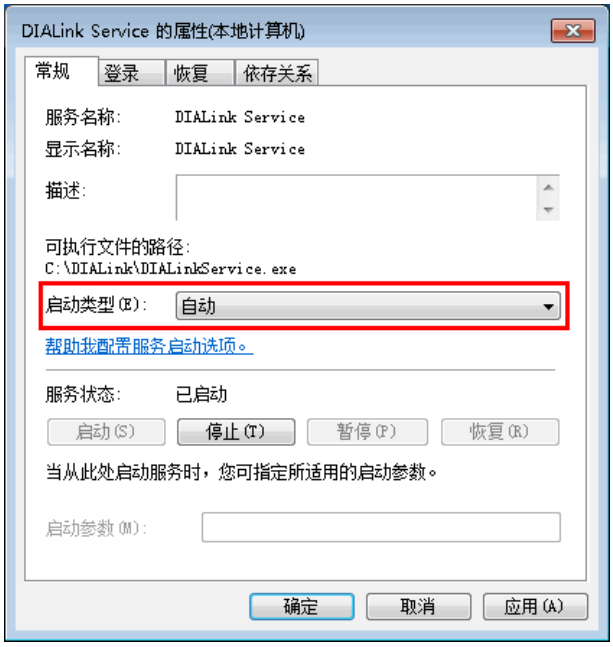


步骤二：选择❶【服务】分页，寻找名称为❷【DIALink Service】的服务。

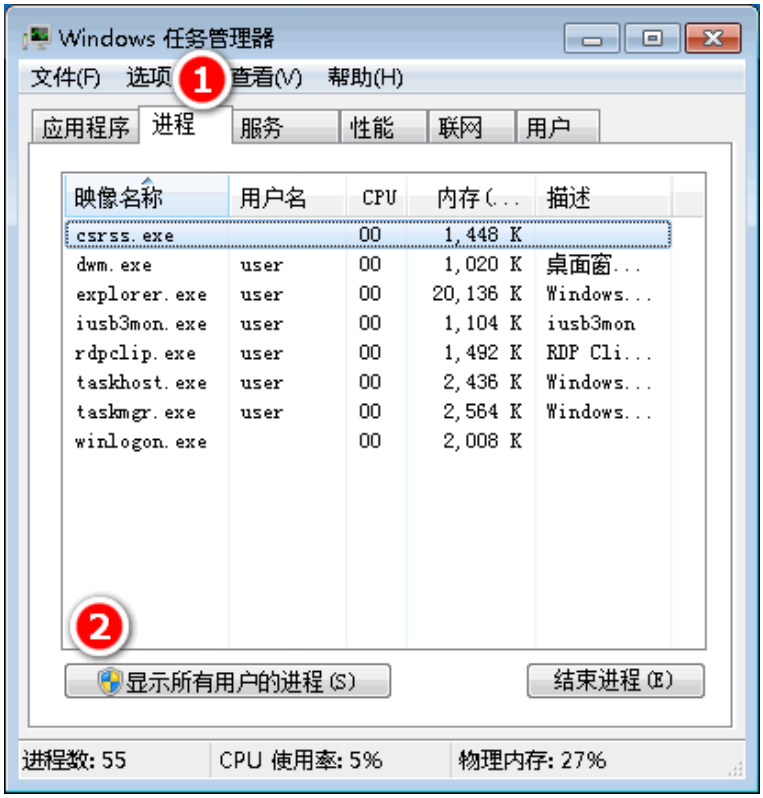


步骤三：若步骤二【DIALink Service】状态为【已停止】，单击窗口右下方【服务】(上图)，寻找名称为❶【DIALink Service】的服务，并单击❷【启动】，同时确认启动类型是否为【自动】；若非【自动】，点击右键 > 【内容】 > 启动类型选择【自动】 > 单击【确定】完成设定。

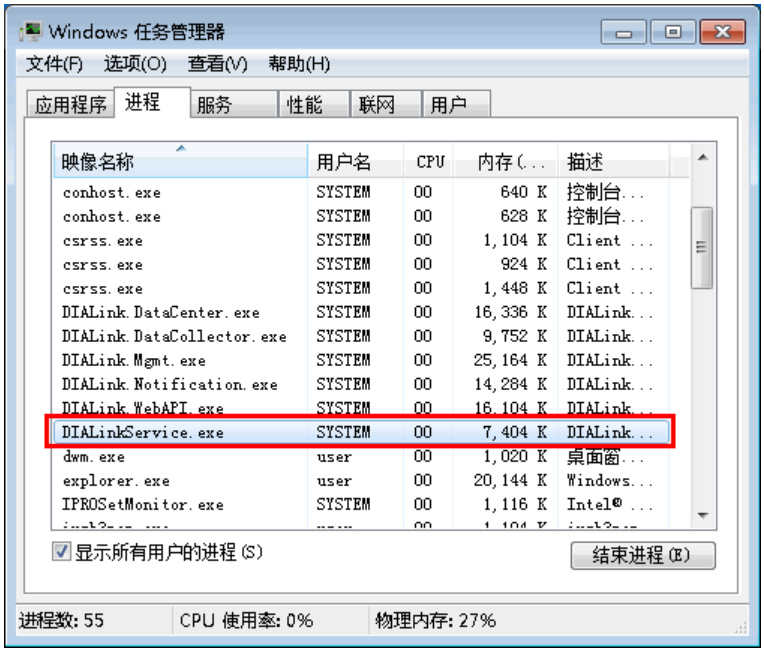




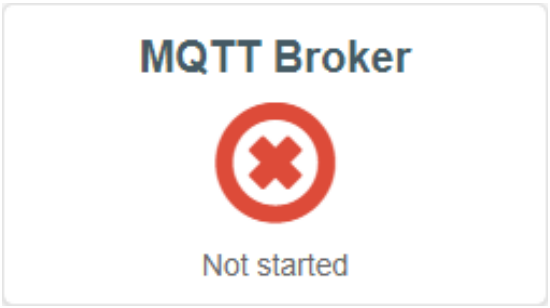
步骤四：回到 Windows 任务管理器，选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。



步骤五：确认是否有【DIALinkService.exe】的进程，若无此进程，重新启动主机。

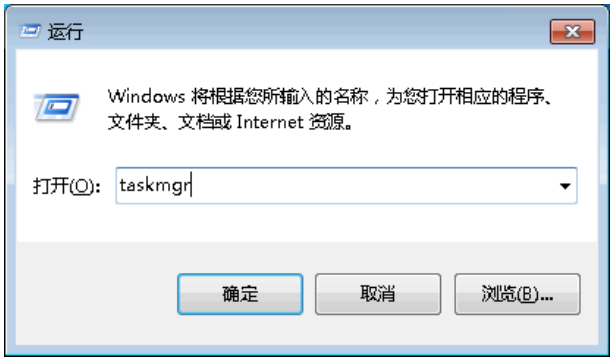


5.1.2 MQTT 服务状态显示 Not Started

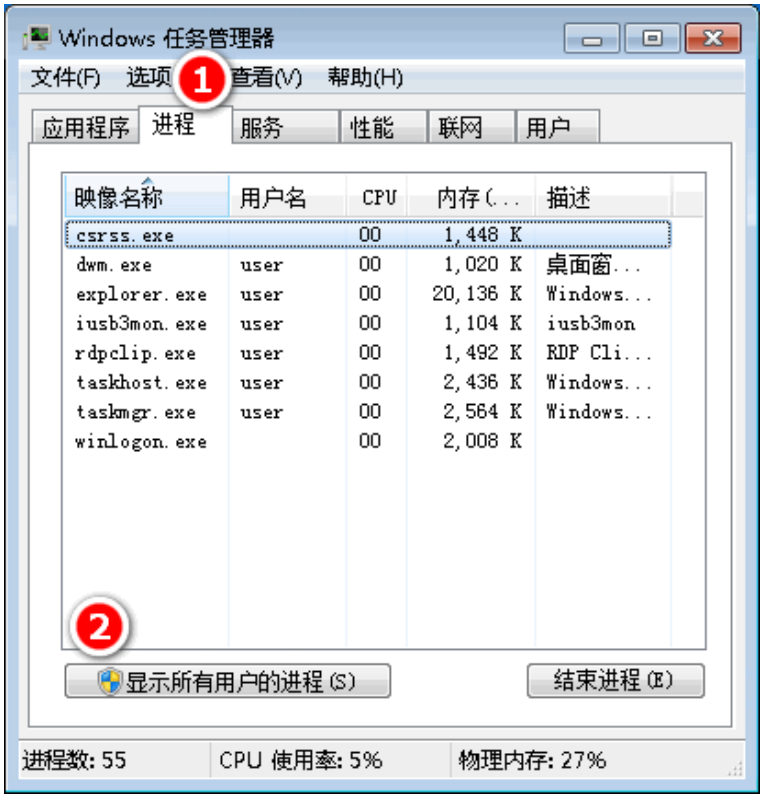


- **处理程序：确认任务管理器内的 MQTT 程序是否启动**

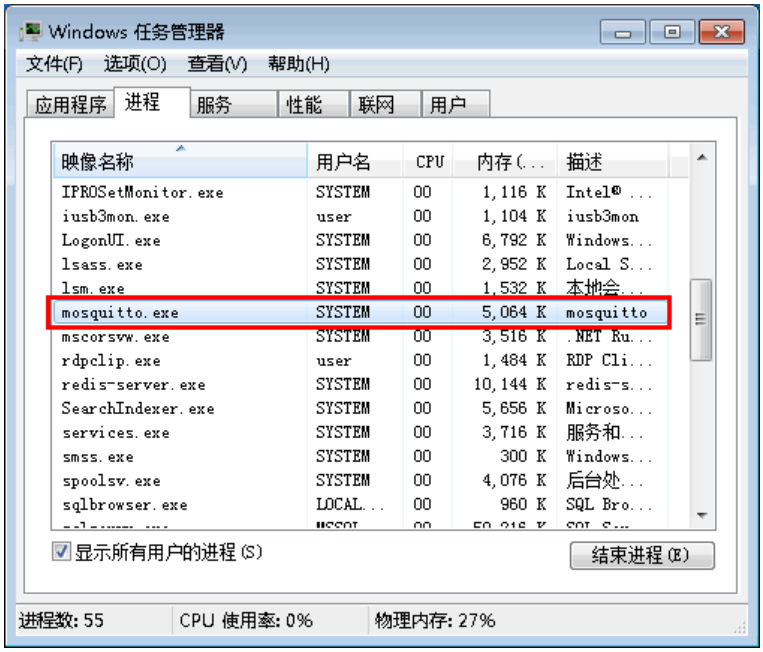
步骤一：按  键 + R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。



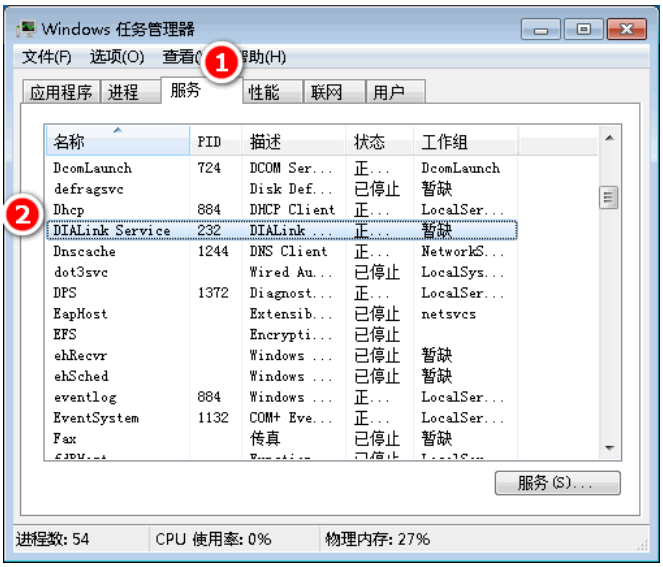
步骤二：选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。



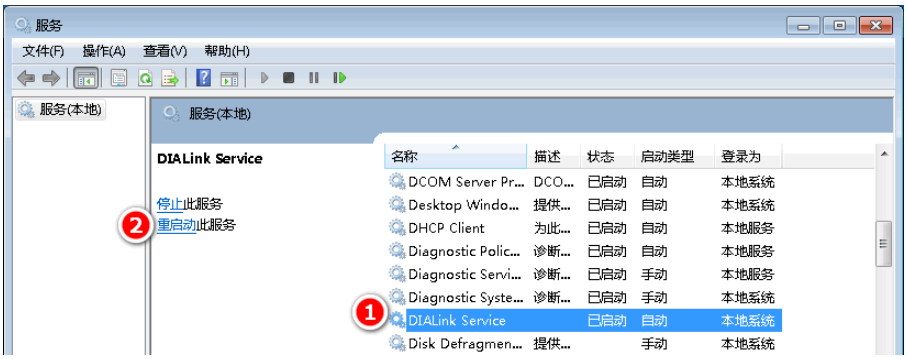
步骤三：确认是否有【mosquitto.exe】的进程，若无此进程，重新启动【DIALink Service】。



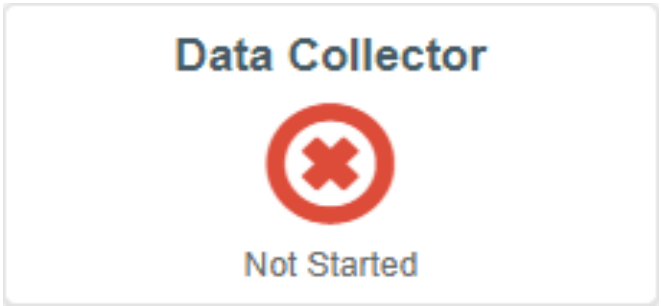
步骤四：选择❶【服务】分页，寻找名称为❷【DIALink Service】的服务，并单击窗口右下方【服务】。



步骤五：寻找名称为❶【DIALink Service】的服务，并单击❷【重新启动】，并再次确认是否有【mosquitto.exe】的进程。

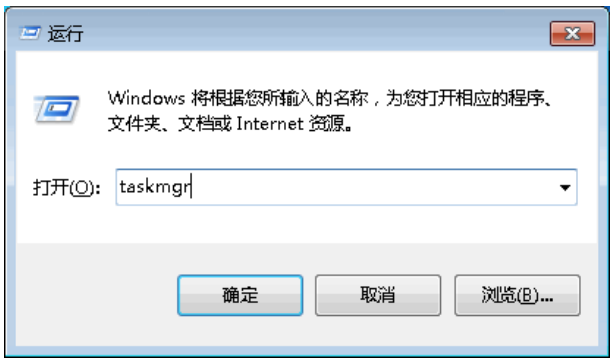


5.1.3 Data Collector 服务状态显示 Not Started

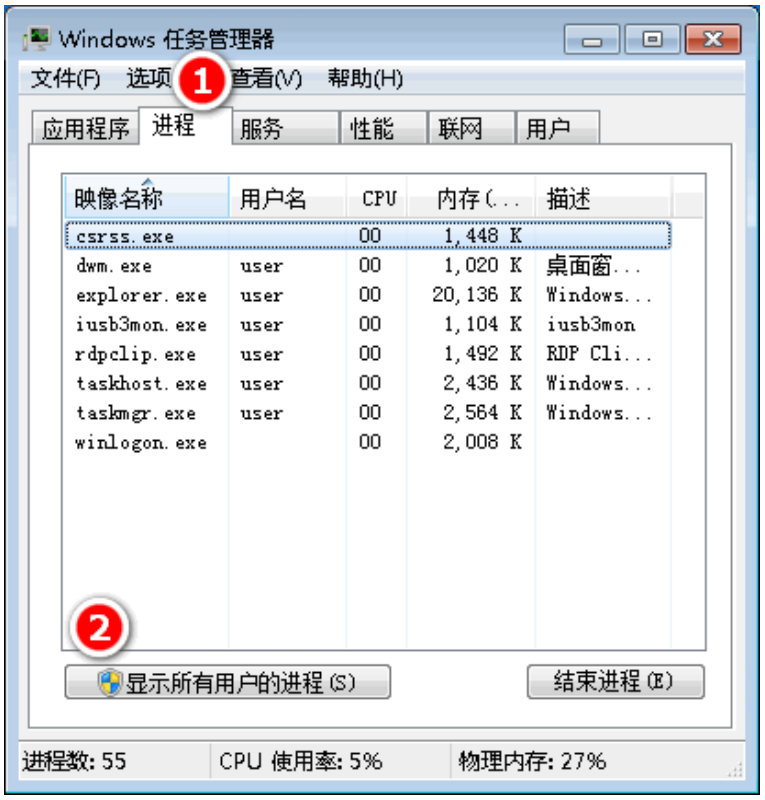


- **处理程序 1：确认任务管理器内的 DataCollector 程序是否启动**

步骤一：按 键 + R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。

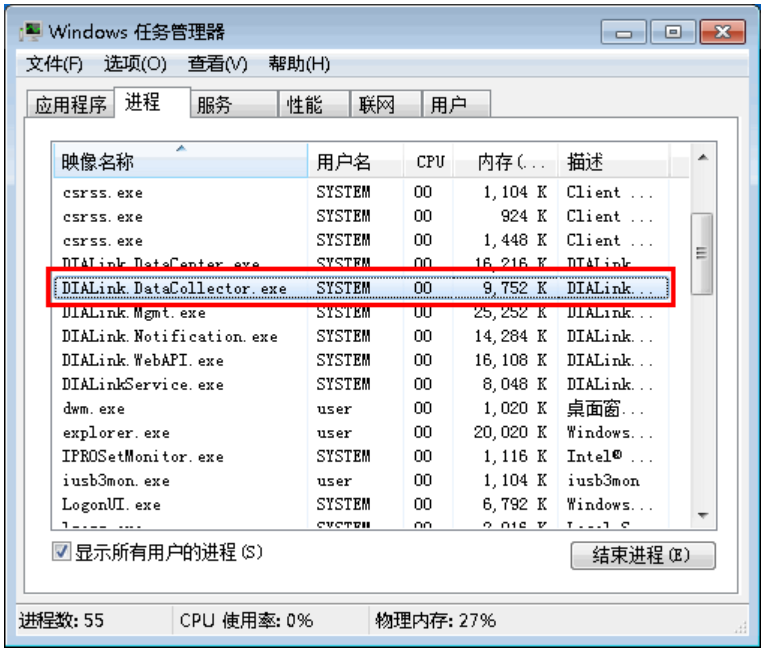


步骤二：选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。

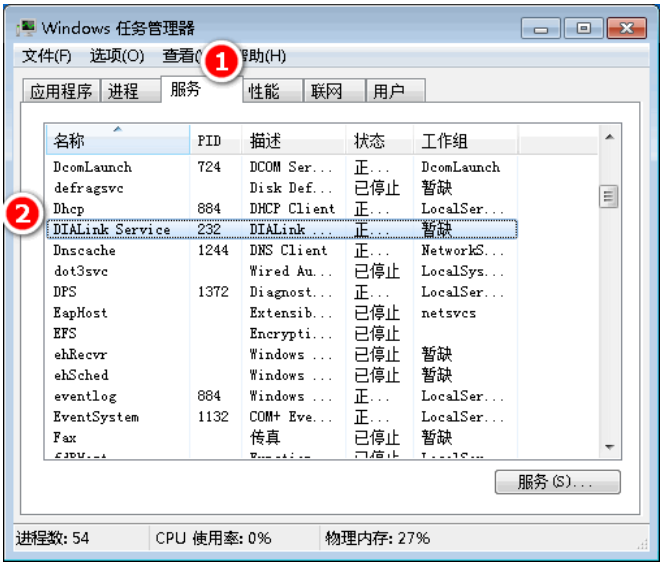


5

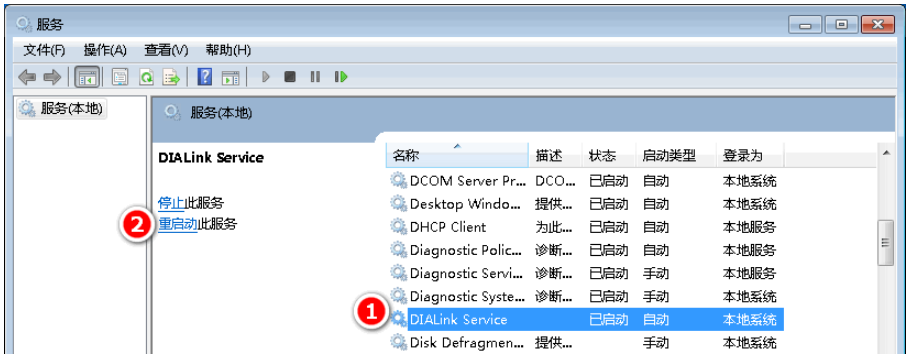
步骤三 :确认是否有【DIALink.DataCollector.exe】的进程 ,若无此进程 ,重新启动【DIALink Service】。



步骤四：选择❶【服务】分页，寻找名称为❷【DIALink Service】的服务，并单击窗口右下方【服务】。



步骤五：寻找名称为❶【DIALink Service】服务，并单击❷重新启动，并再次确认是否有【DIALink.DataCollector.exe】进程，若【Data Collector】服务状态仍然显示 Not Started，进行处理程序 2。

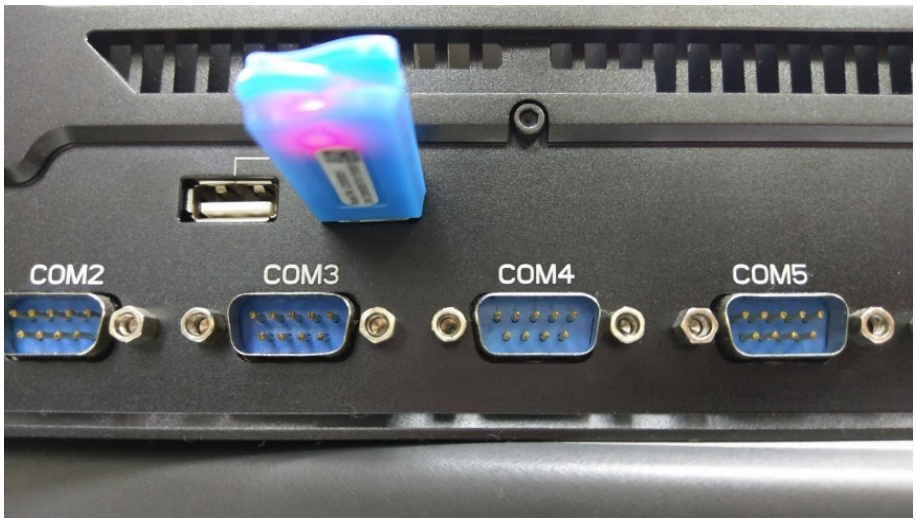


- **处理程序 2：确认 DIALink 授权**

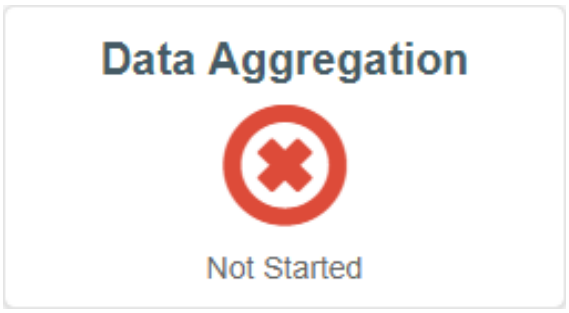
步骤一：查看【DIALink 信息】，登入 DIALink，单击右上方，选择【关于 DIALink】查看授权状态，是否显示设备 / 点位数授权数量。



步骤二：若未显示授权数量的设备台数与点位点数，请确认 USB 加密锁是否正确插入主机并亮红灯。

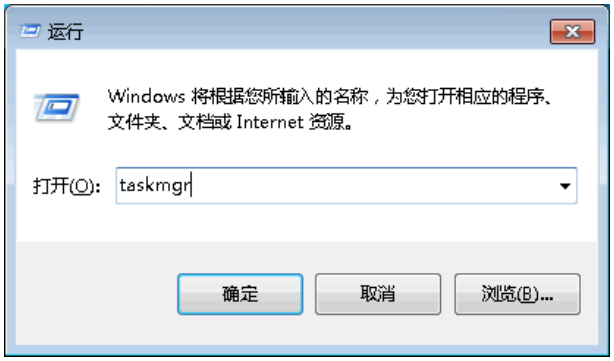


5.1.4 Data Aggregation 服务状态显示 Not Started

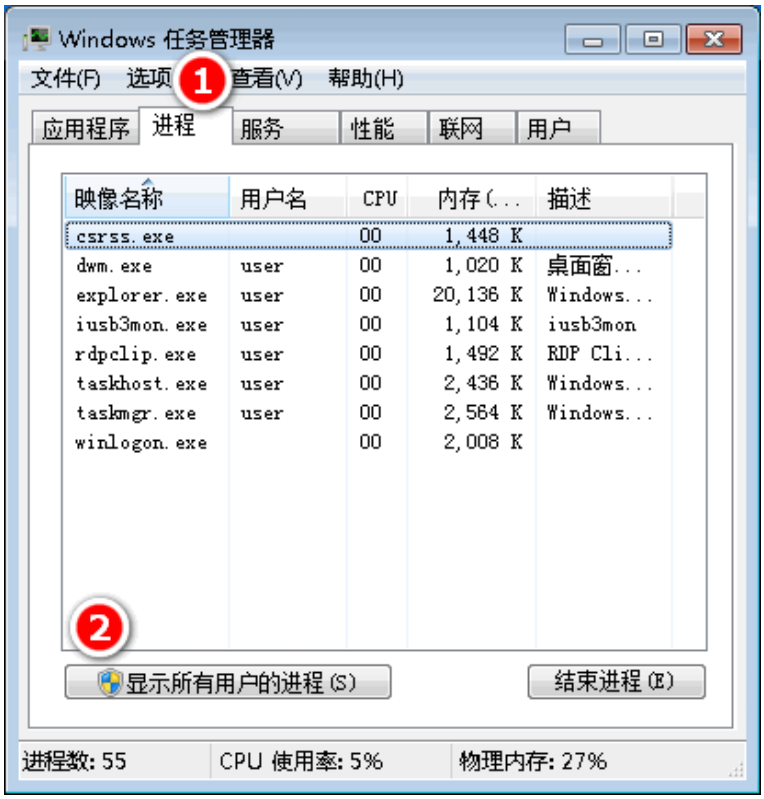


- **处理程序：确认任务管理器内的 DataCenter 程序是否启动**

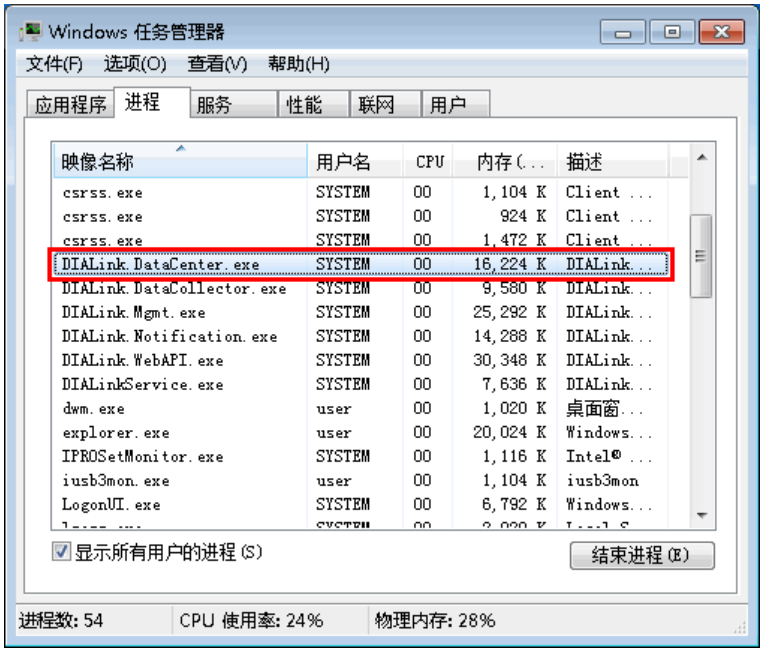
步骤一：按 键 + R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。



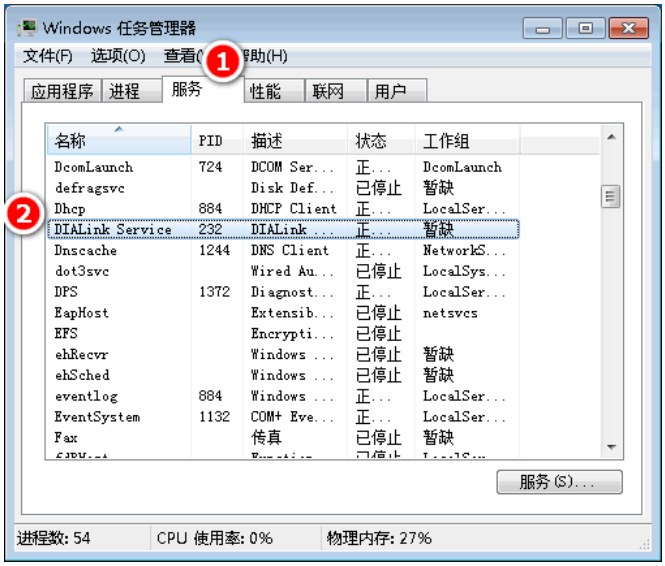
步骤二：选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。



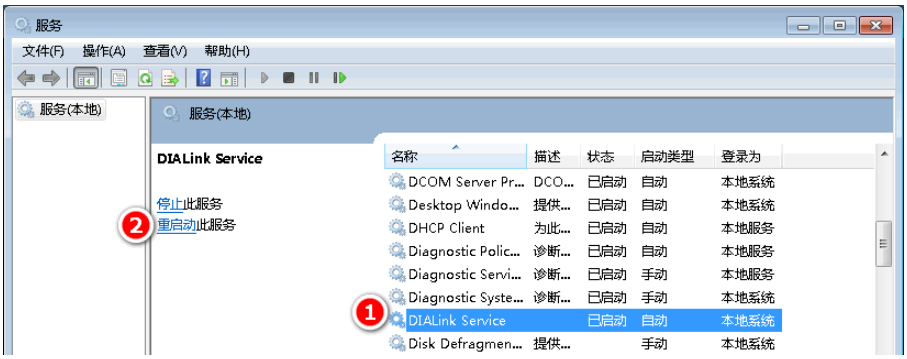
步骤三：确认是否有【DIALink.DataCenter.exe】进程，若无此进程，重新启动【DIALink Service】。



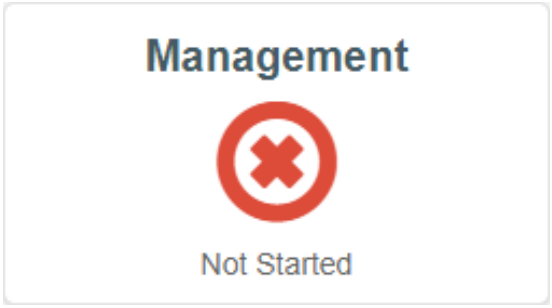
步骤四：选择❶【服务】分页，寻找名称为❷【DIALink Service】服务，并单击窗口右下方【服务】。



步骤五：寻找名称为❶【DIALink Service】服务，单击❷【重新启动】，并再次确认是否有【DIALink.DataCenter.exe】进程。

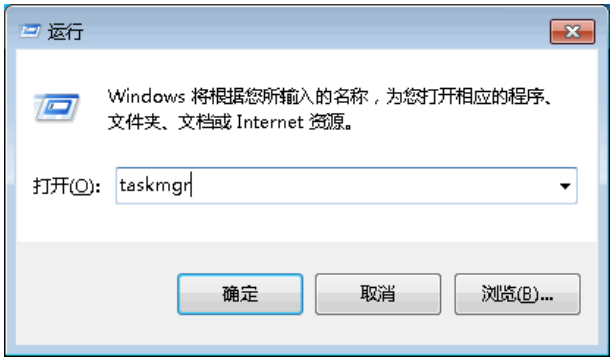


5.1.5 Management 服务状态显示 Not Started

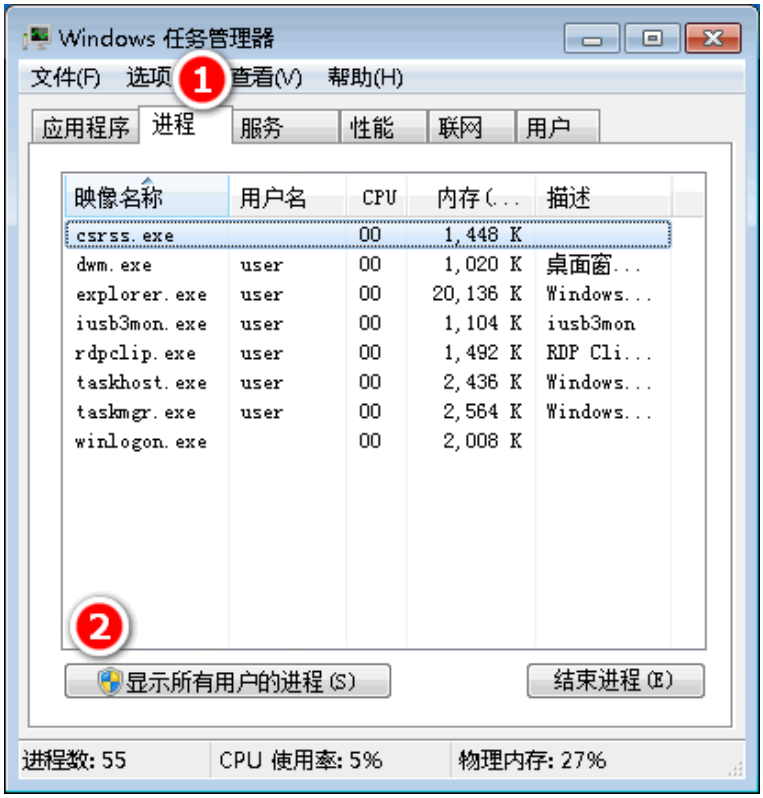


- **处理程序：确认任务管理器内的 DIALink.Mgmt.exe 程序是否启动**

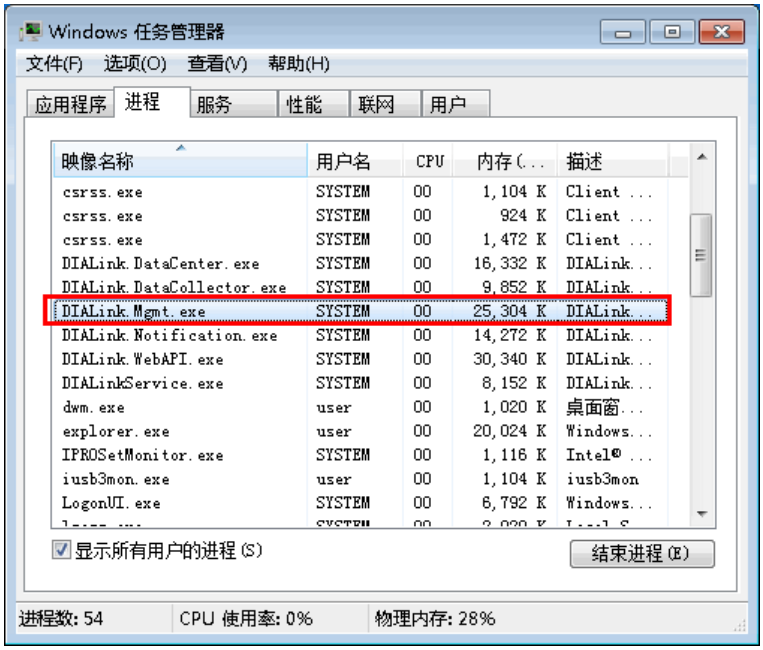
步骤一：按  键 + R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。



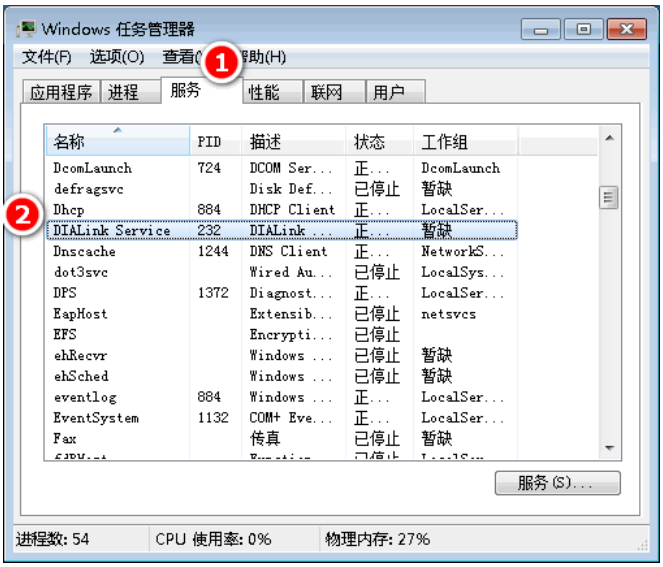
步骤二：选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。



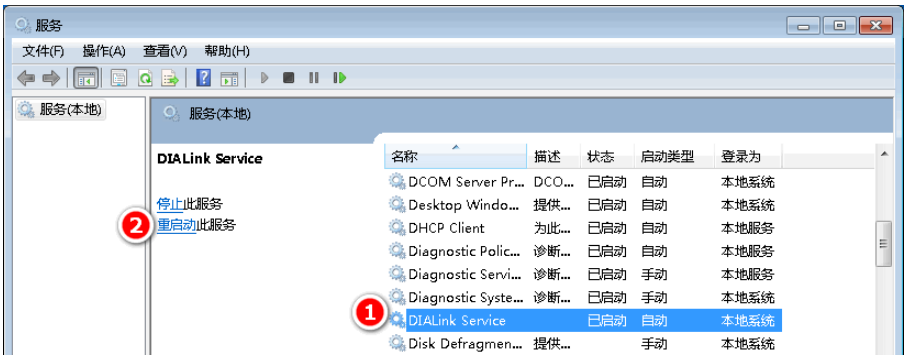
步骤三：确认是否有【DIALink.Mgmt.exe】进程，若无此进程，重新启动【DIALink Service】。



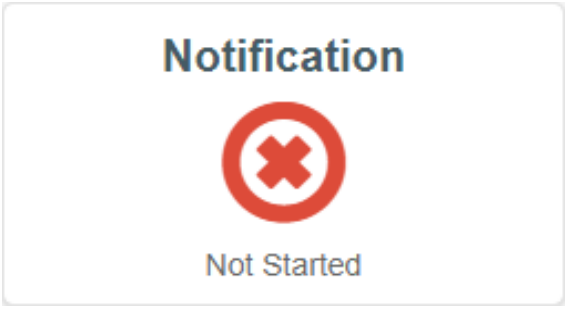
步骤四：选择❶【服务】分页，寻找名称为❷【DIALink Service】服务后，单击窗口右下方【服务】。



步骤五：寻找名称为❶【DIALink Service】服务，单击❷【重新启动】，并再次确认是否有【DIALink.Mgmt.exe】进程。

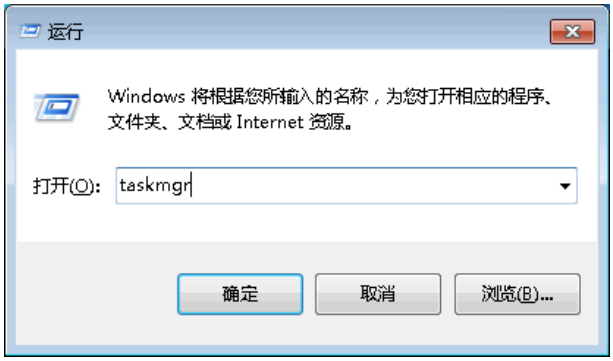


5.1.6 Notification 服务状态显示 Not Started

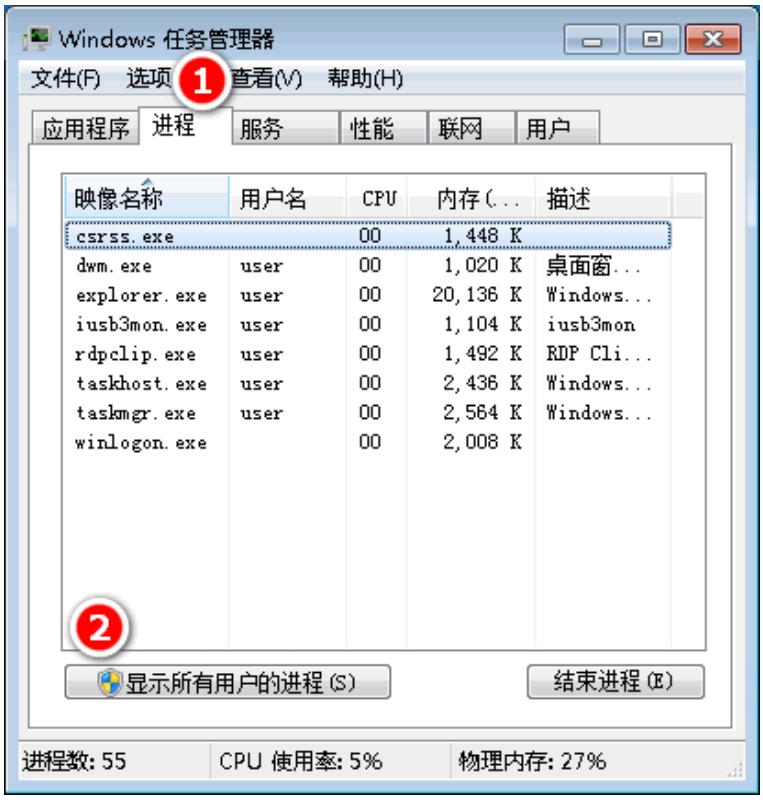


- **处理程序：确认任务管理器内的 Notification 程序是否启动**

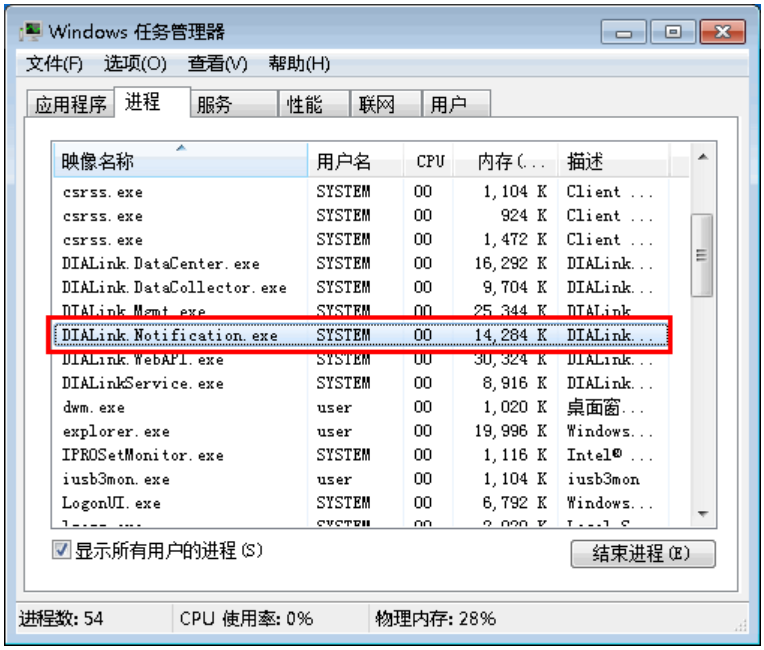
步骤一：按 键 +R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。



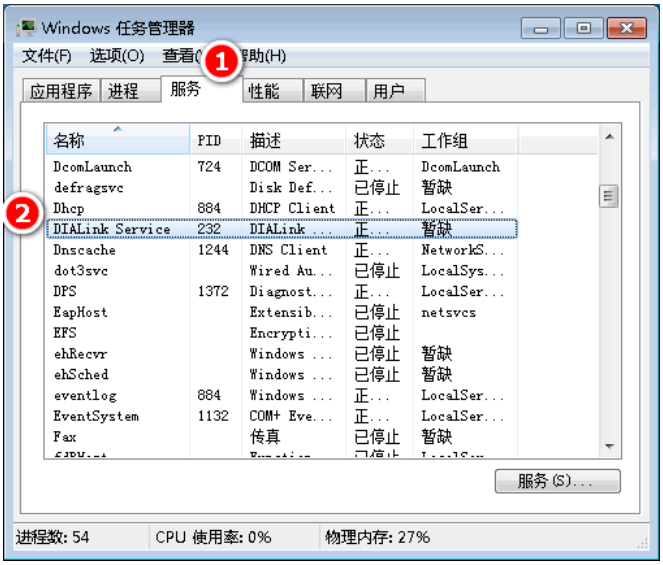
步骤二：选择❶【进程】分页，单击❷【显示所有用户的进程(S)】。



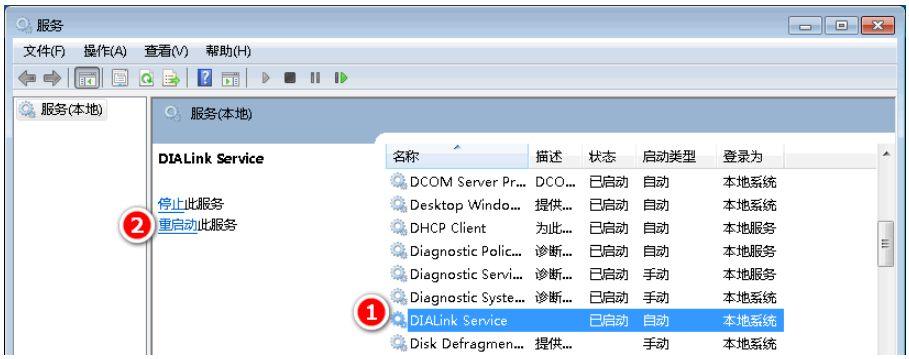
步骤三：确认是否有【DIALink.Notification.exe】进程，若无此进程，重新启动【DIALink Service】。



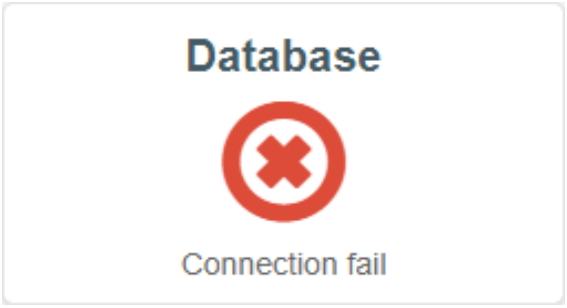
步骤四：选择❶【服务】分页，寻找❷名称为【DIALink Service】服务后，单击窗口右下方【服务】。



步骤五：寻找名称为❶【DIALink Service】服务，单击❷【重新启动】，并再次确认是否有【DIALink.Notification.exe】进程。

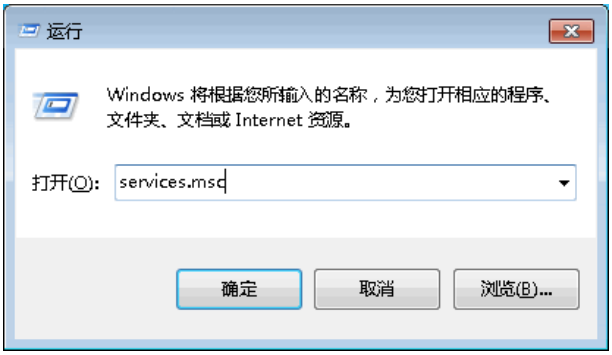


5.1.7 Database 服务状态显示 Not Started

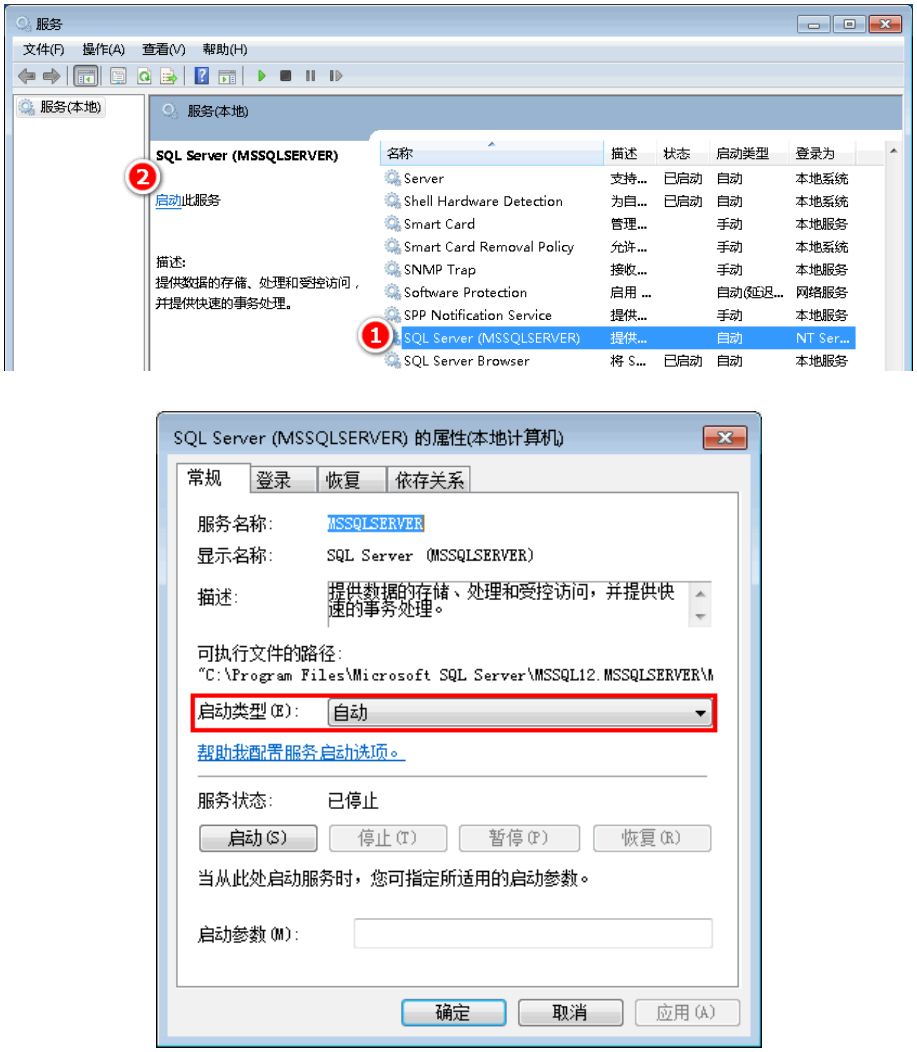


- **处理程序 1：确认任务管理器内的 SQL 服务是否启动**

步骤一：按  键 + R 或【开始】 > 【执行】，输入 taskmgr 进入 Windows 任务管理器。



步骤二：寻找名称为❶【SQL Server（MSSQLSERVER）】服务，若状态不为【已启动】，单击【启动】，同时确认启动类型是否为【自动】；若非【自动】，点击右键 > 【内容】 > 启动类型选择【自动】 > 单击【确定】完成设定（下图）。



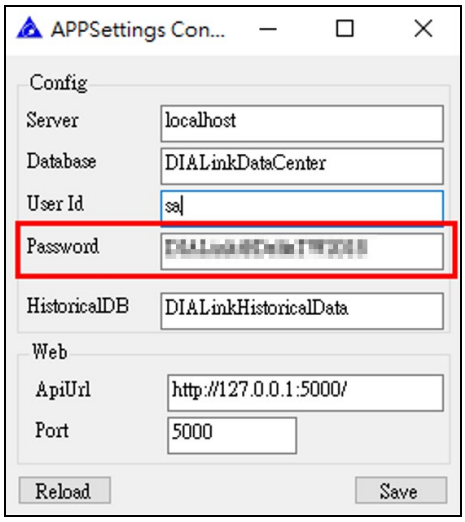
- **处理程序 2**：确认硬盘是否有足够的空间存放 SQL 数据



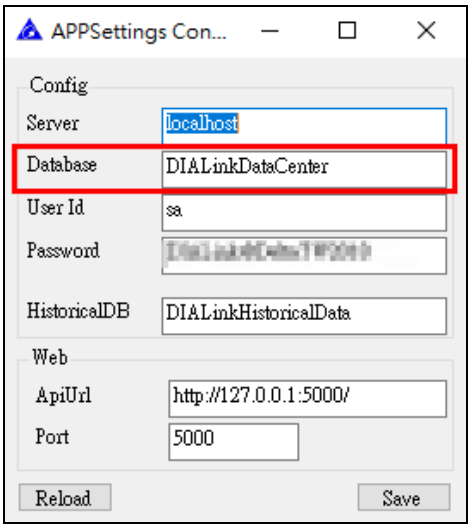
5.1.8 数据库依旧无法正常运作

以下为常见的数据库无法正常运作的原因：

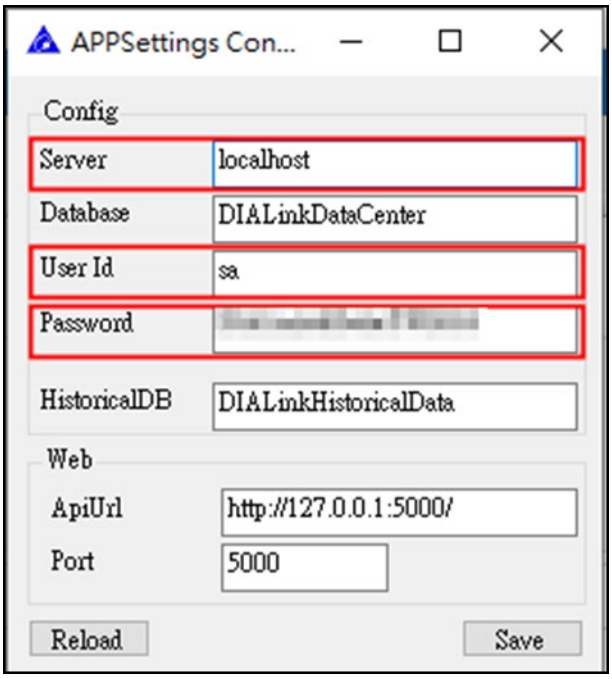
1. 硬盘空间最少清出 1G，比较能维持正常运行。
2. 数据库 sa 账号的密码被修改，在一个全新的 IPC 上安装好的 DIALink 会建立并使用默认的 sa 账号与密码，请客户先确认 DIALink 的默认 sa 账号与密码是否被其他公司员工改过，若有则改回来即可正常联机运作 若因特殊原因不能改回 sa 密码 则要通过 APPSettingsConfigTool.exe 修改 DIALink 的参数设定中的密码字段，将这个密码字段改成用户所用的数据库 sa 密码。这个动作仅是告知 DIALink 数据库联机参数有变更，并不会影响用户所用的数据库 sa 密码。



3. 若卸载 SQL Server，重新安装一次，会导致 SQL Server 内有（DIALinkDataCenter ）相同名称的数据库，进入数据库修改名称（或删除）即可。



4. 若客户想使用自己现有的数据库 也限定只能使用 SQL Server 且客户的数据库必须设定为 SQL Server 服务器验证方式，且具备 sa 账号及密码，或者其他密码及具备 db_owner 权限的使用者账号。请至 C:\DIALink 执行 DIALink.APPSettingsConfigTool.exe，输入客户的 SQL Server 的地址、User ID、Password。最后请务必重启 DIALink 服务，变更的设定值才会生效。



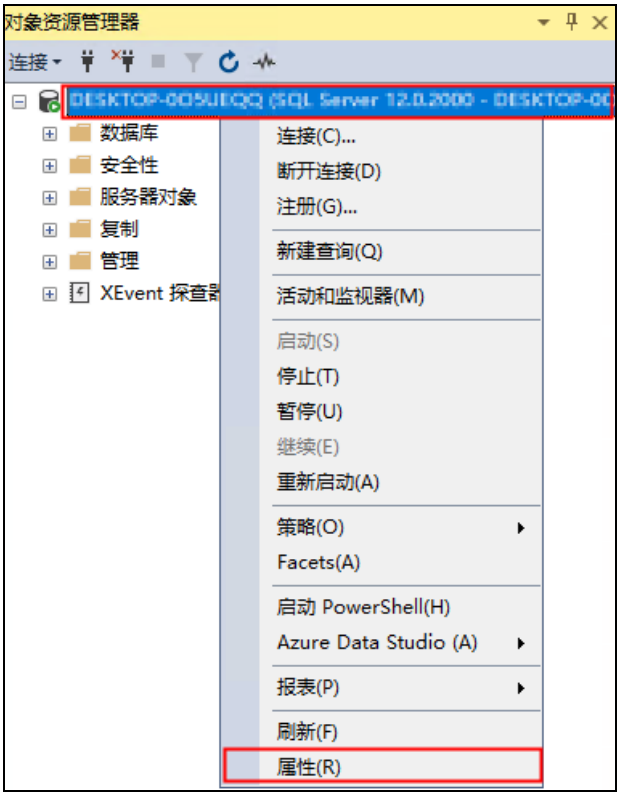
5. 若客户使用自己的数据库，且原本为 Windows 认证，可通过以下更改设定步骤变成具备 SQL Server 服务器验证方式：

   步骤一：请先下载 SQL Server Management Studio，此软件简称 SSMS，请从微软官方免费下载安装。

   步骤二：请以 Windows 验证方式先登入。

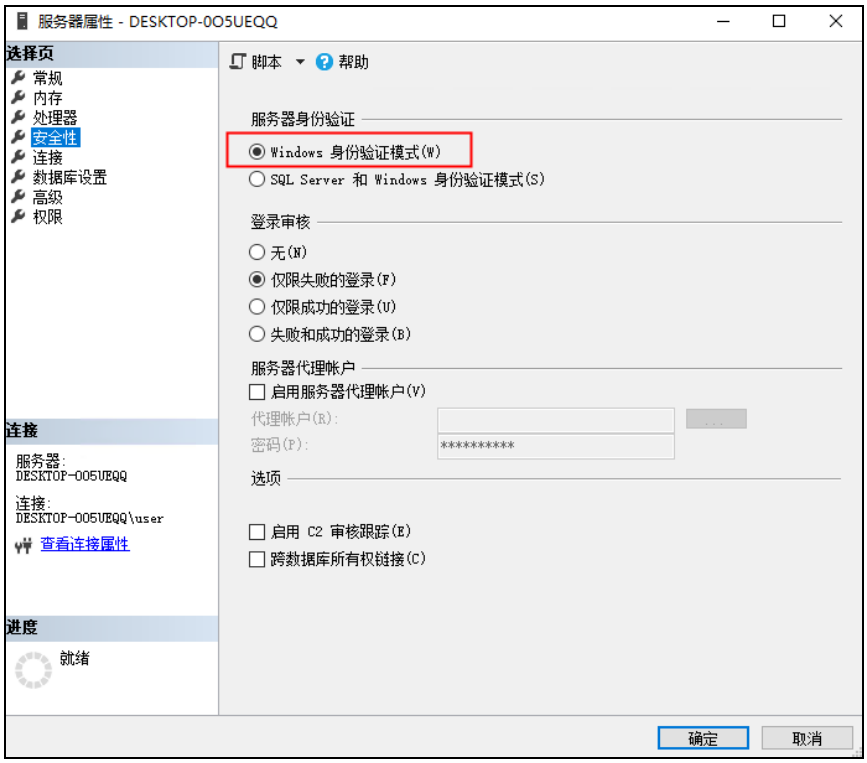


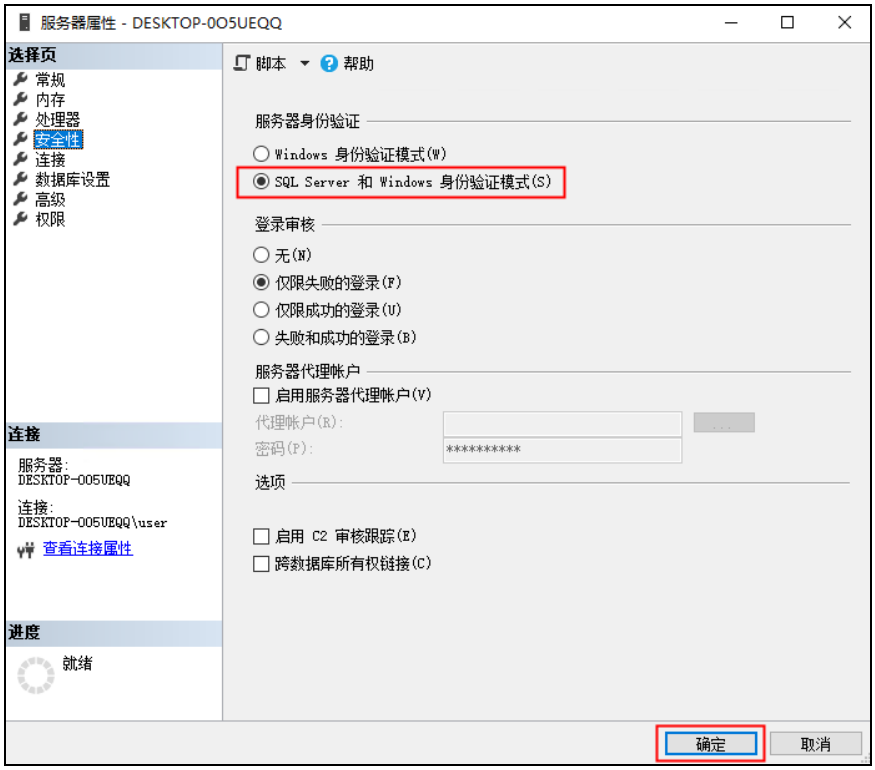
   步骤三：在对象总管中，以鼠标右键单击服务器，然后单击【属性】。



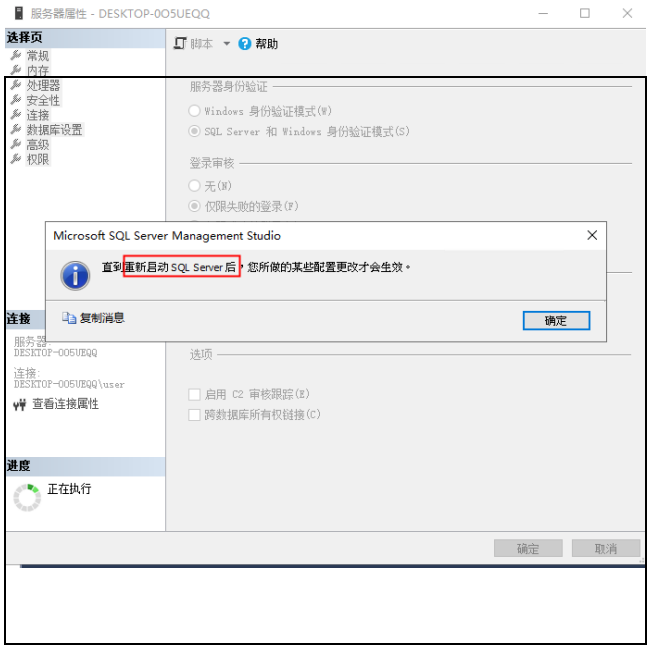
步骤四：在【安全性】页面上的【服务器验证】中，从【Windows 验证模式】改为【SQL Server 及 Windows 验证模式】。

步骤五：在对话框中，单击【确定】以确认需要重新启动【SQL Server】。

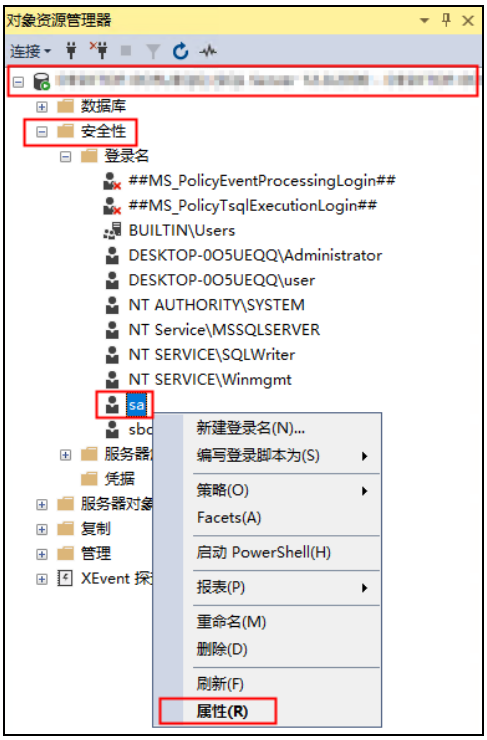




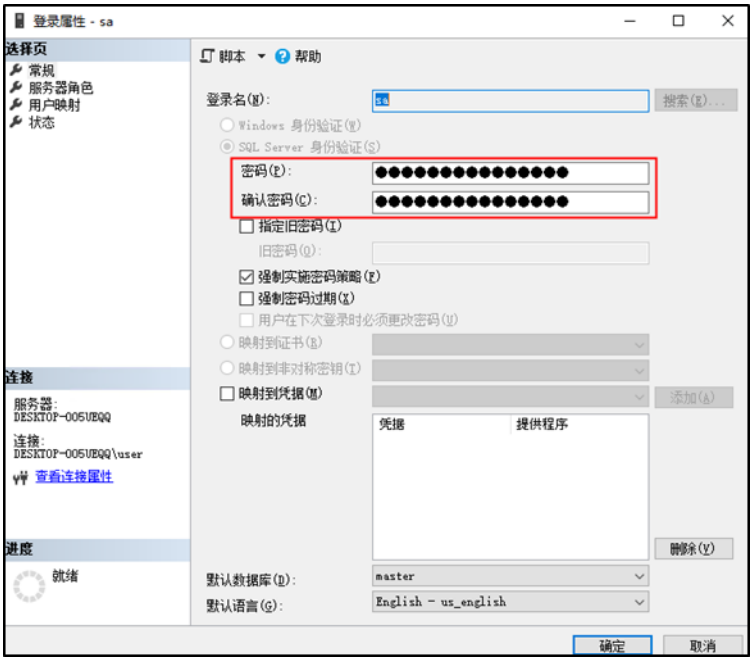
步骤六：在对象总管中，以鼠标右键单击服务器，然后单击【重新启动】。如果【SQL Server Agent】正在执行，也必须将它重新启动。



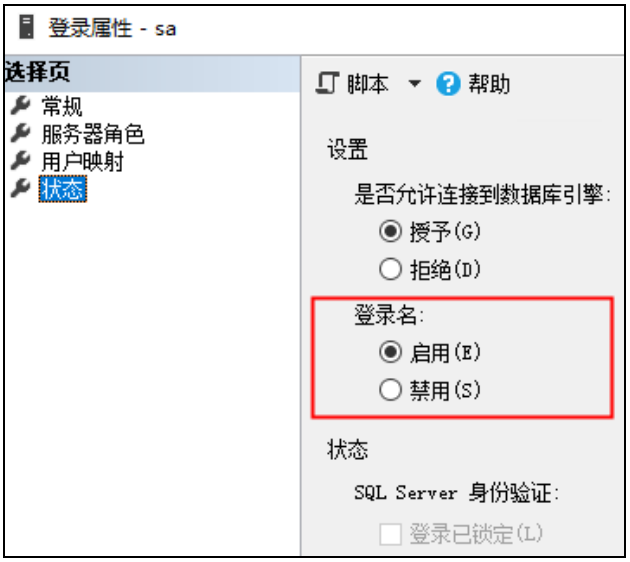
步骤七： 在对象总管中，展开【安全性】和【登入】，并以鼠标右键单击【 sa 】，然后按 【属性】。



步骤八：在【一般】页面上，请输入密码，此密码需要跟 DIALink 的数据库密码设定一样，可以通过执行 DIALink.APPSettingsConfigTool.exe 的 Password 来确认目前密码。如果客户有特殊需求，必须设定为特定的密码，则在此输入密码后，执行 DIALink.APPSettingsConfigTool.exe 将 Password 修改，最后请务必重启 DIALink 服务，变更的设定值才会生效。



步骤九：在【状态】页面的【登入】区段中确认（或修改）为【已启用】，且【联机到数据库引擎的权限】确认为【授与】，然后按【确定】即可完成验证方式变更。



5.1.9 插上硬件 KEY 发生异常，仅能使用试用版

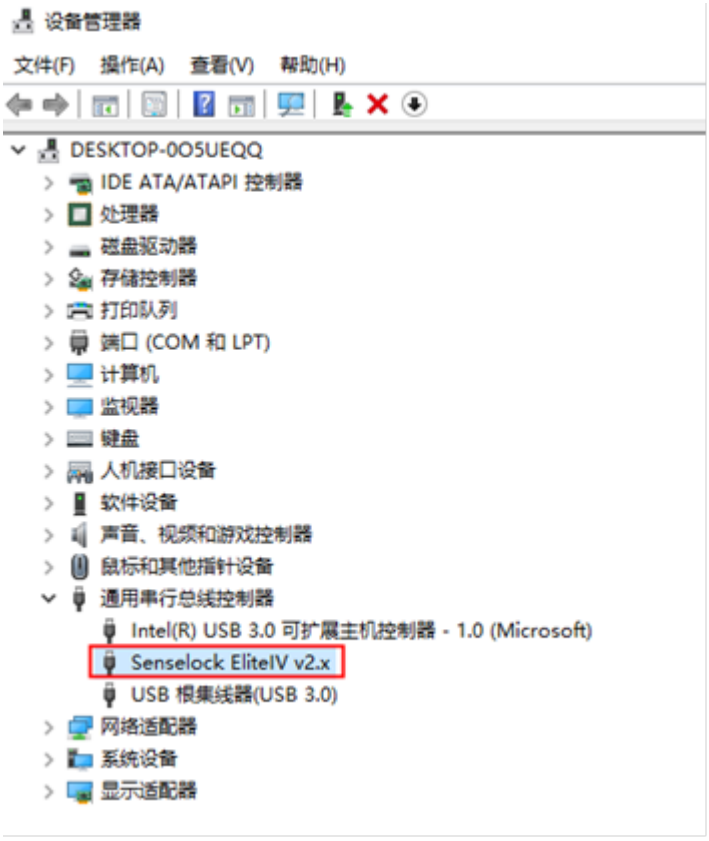
1. [Q] 插上硬件 KEY 却无法看到正确授权，仅能使用试用版。

   [A] 通常会发生多半是因为 USB 的插孔没有正常运作，请先换上其他 USB 装置，比方鼠标、键盘等，确认该 USB 孔是否正常运作，若插孔确认有问题，换其他插孔即可。不论是 USB 孔有问题找不到 dongle 装置还是 dongle driver 没安装好，应该会显示”Cannot find any dongle”。



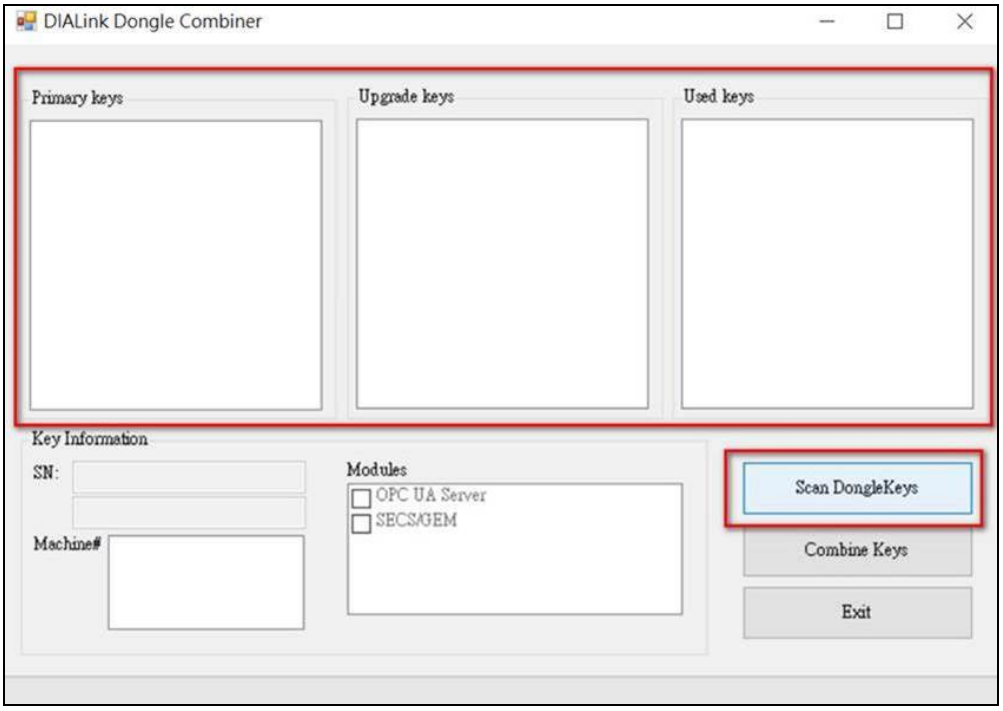
2. [Q] 确定 USB 孔正常，硬件 KEY 插上仅能使用试用版。

   [A] 可能因为运送过程的问题造成硬件 KEY 内的信息无法读取辨识，在设备管理器中虽然可看到 Senselock 名称的装置，但里头数据已经无法读出，请跟代理商联络，我们会尽快更换一支硬件 KEY 给您。

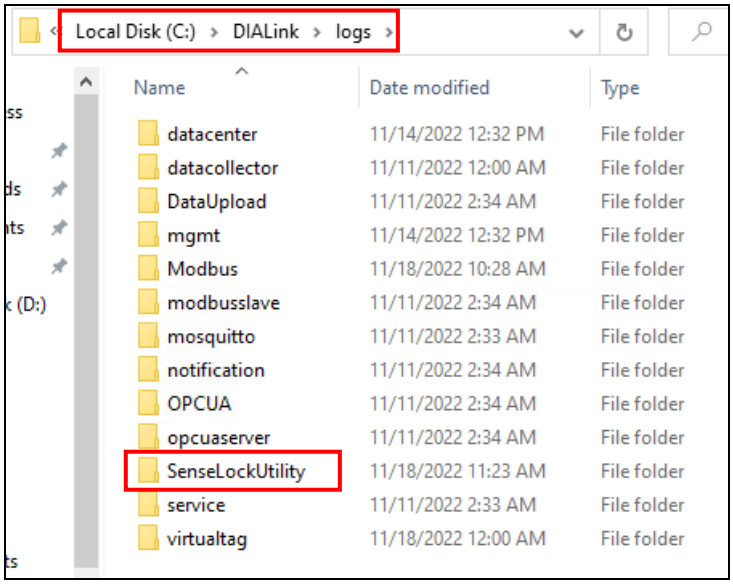


3. [Q] 硬件 KEY 可读取，里头数据也可读出，但内容错误，不是序号。

[A] 请执行 CombinDongleKesForm.exe，如果 SCAN 后红框内三个(Primary keys, Upgrade keys, Used keys)都是空白，就表示这只 dongle 里面不是我们 link 的资料，读不到内容，请跟代理商联络，我们会尽快更换一支硬件 KEY 给您。



4. [Q] 硬件 KEY 可读取，里头数据也可读出，但内容序号跟硬件 KEY 上卷标的序号不同。
   [A] 请跟代理商联络，我们会尽快更换一支硬件 KEY 给您。
5. [Q] 上述情况都没有发生，但插上硬件 KEY 仍无法看到正确授权，仅能使用试用版。
   [A] 请跟代理商联络，另外请把整个 logs 下面 SenseLockUtility **所有 log** 取回来，方便查看问题，请将目录下所有文件压缩成 ZIP 档，不要仅压缩一日或两三日的文件。



5.1.10 无法联机到 CNC 设备

1. 通常遇到最多的问题是客户并未开通 CNC 控制器的数据采集授权（DNC 授权），各家 CNC 制造商开通授权方式不同，以海德汉来说，要开通 18N 的模块，并且海德汉 640 联机需要设定密码；FANUC 则是购买后安装 FOCAS 模块，并开启 DNC 授权。

2. 网络设定错误也是造成无法联机的常见原因，网络设定请参照操作手册及贵公司的 IT 部门规定。至少要 PING 到设备 IP，且 Telnet 得到设备所使用的 Port。（各品牌 CNC 所使用的 Port 不相同，请自行参阅各家设备手册）。

A

附录A Modbus Slave 功能

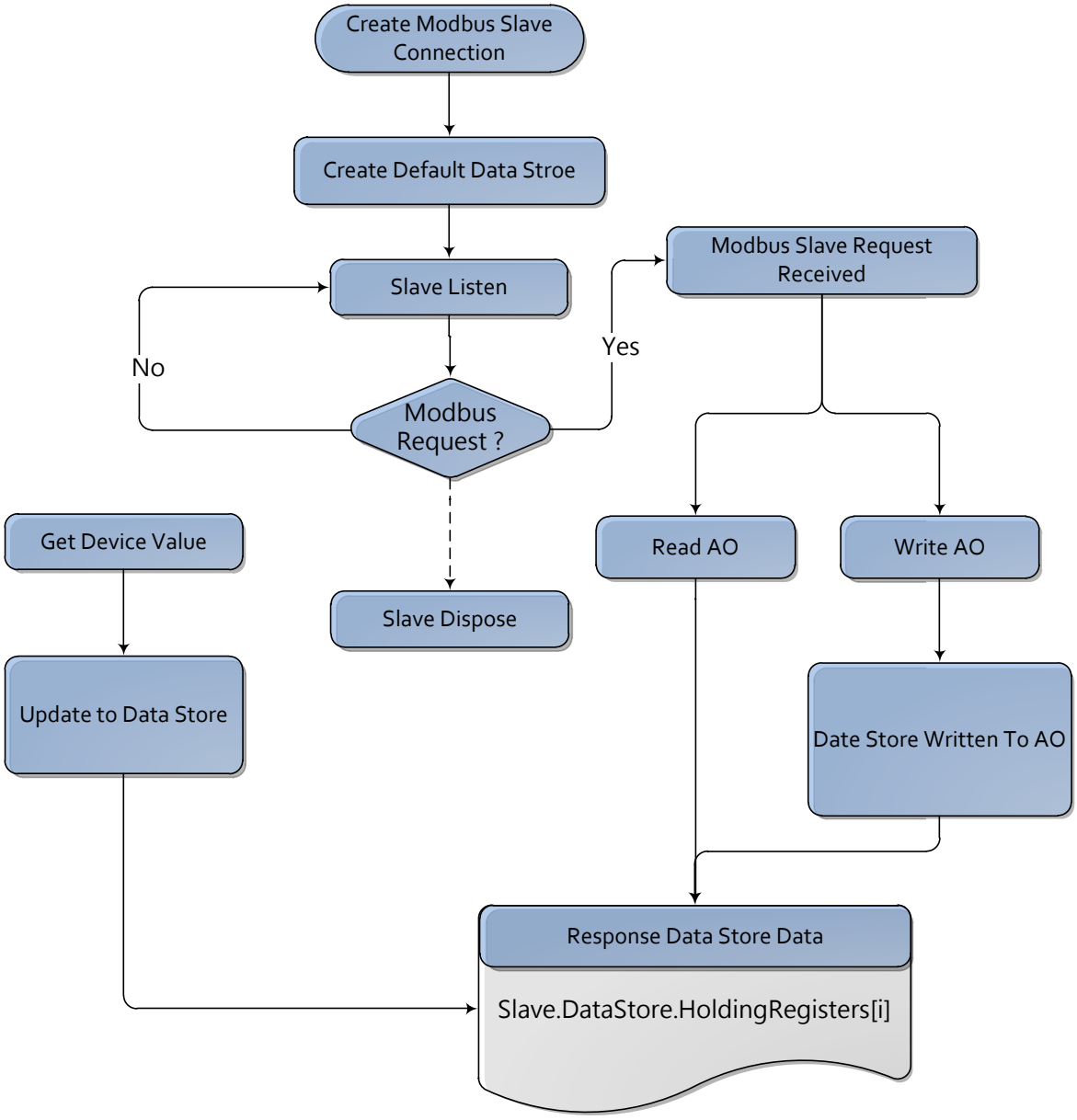
目录

A.1 概述 A-2
A.2 规格说明 A-3

A.1 概述

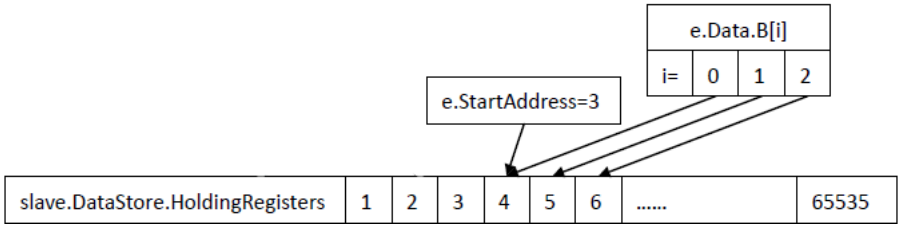
Modbus 是一种需求-响应协议，采用主从架构实作而成。主从架构的通讯作业会成双成对的出现，必须有个装置启动需求并等候响应，也就是说，这个启动需求的装置（主要装置）会负责启动每一次互动。一般而言，主要装置（Master）会是人机接口（HMI）或监控与数据获取（SCADA）系统，附属装置（Slave）则可以是传感器、程序化逻辑控制器（PLC）、程序化自动控制器（PAC）或数据服务应用程序。

DIALink Modbus Slave 则是一个数据服务应用程序，仅支持通过 TCP/IP 沟通及寄存器 AO（Holding Register）读取方式，提供第三方平台取得系统收集到的装置数据。目前暂无写入寄存器功能。



A.2 规格说明

DIALink Modbus Slave 通过 Holding Register 寄存器数组 1～65535 存入装置数据。每次读取寄存器长度限制为 127。



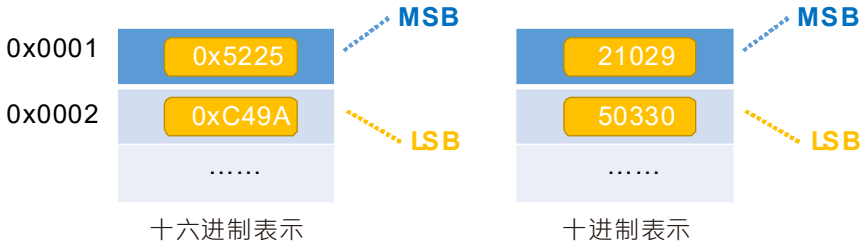
数据类型：目前仅支持数值及 ASCII 字符串。

数值数据：一律以 Float（占用 2 个寄存器数组空间）类型，以 Little-endian 排列方式存取。

字符串数据：仅支持字符为 ASCII 范围 X20~X7F，以 Little-endian 排列方式存取。

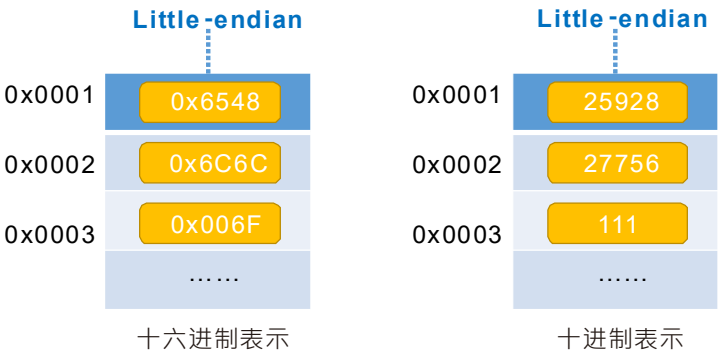
以浮点数-1234.567 为例：

1. 转换成 16 进制（不足 8 位左边补 0）字符串 C49A5225。
2. 以 Little-endian 排列方式转换为 Uint16 储存至寄存器。
3. 浮点数内容会存在误差，通过寄存器存取会有小数点精确度的问题。



以字符串 Hello 为例：

1. 转换成 16 进制（不足 4 的倍数右边补 0）字符串 48656C6C6F00 表示。
2. 根据来源类型，单一数组内容以 Little-endian 排列方式转换为 Uint16 储存至寄存器。



B

附录B IFTTT Line 与微信公众号设定

目录

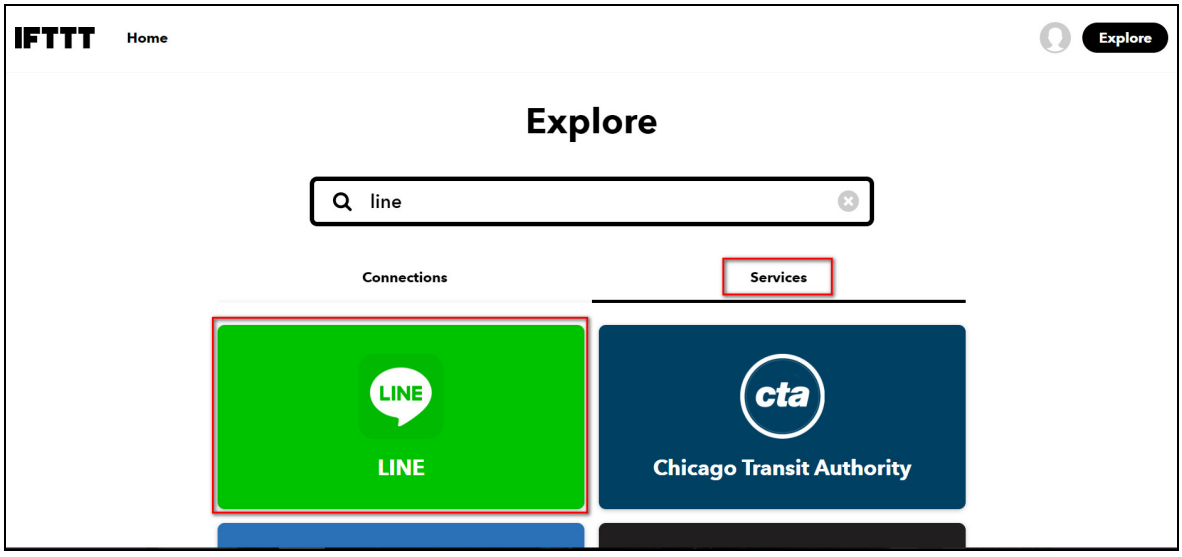
B.1 IFTTT Line 设定..... B-2
B.2 微信公众号申请教学 B-11

B.1 IFTTT Line 设定

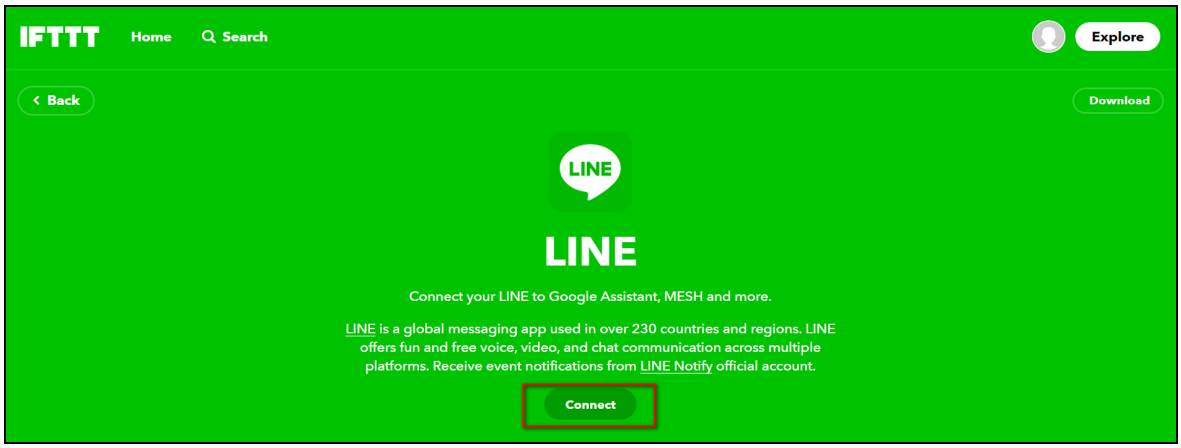
在 IFTTT 里启用 LINE Notify。IFTTT 是个网络服务平台，缩写的意思是 IF（if）T（this）T（then）T（that），顾名思义就是「若做了什么（this）则（then）就执行什么（that）」，通过其它不同平台的条件来决定是否执行下一条命令。即对网络服务通过其它网络服务作出反应。

Step 1：单击连结（https://ifttt.com），进入 IFTTT 网站，单击【Sign up】注册账号并登入，搜索【line】，切换至【Services】，并单击 LINE。

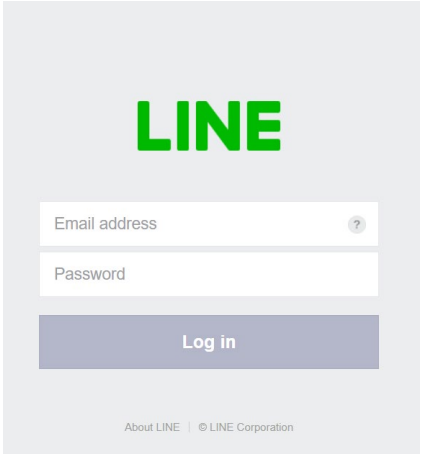


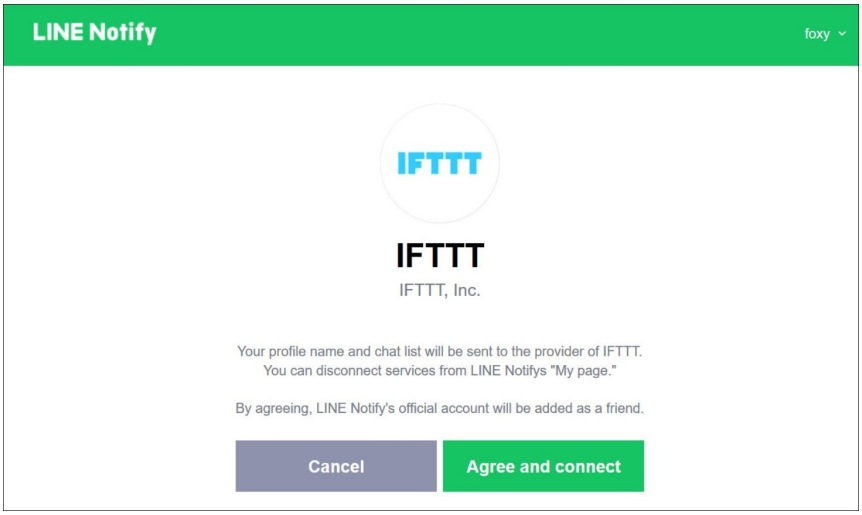


Step 2：单击后，可看到一个 Connect 的按钮，目的在于要链接 LINE 官方所提供的 Notify 通知服务，单击 Connect 就可以进行连结。

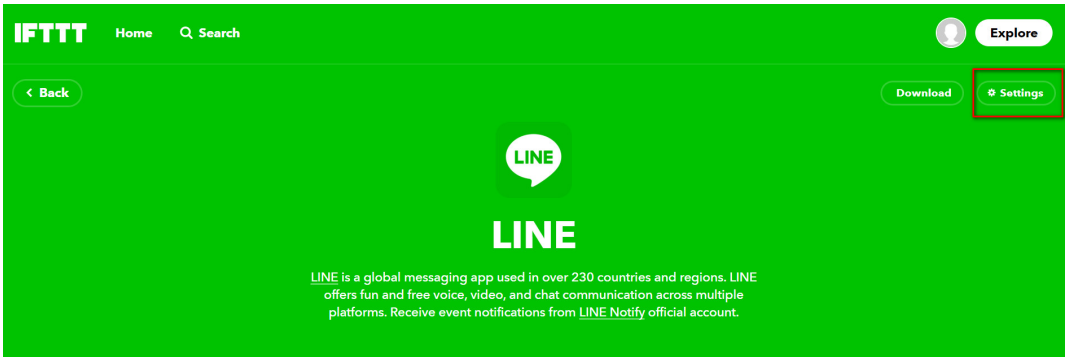


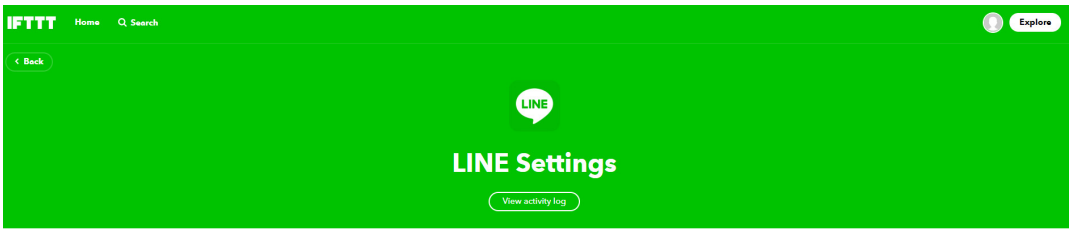
Step 3：单击后，页面会导向 LINE 的网页登入画面，使用自己的 LINE 账号登入，登入后单击【Agree and connect】，同时在你的 LINE 好友里，也会自动新增名为 LINE Notify 的好友，届时就是这个好友会发送通知给你。





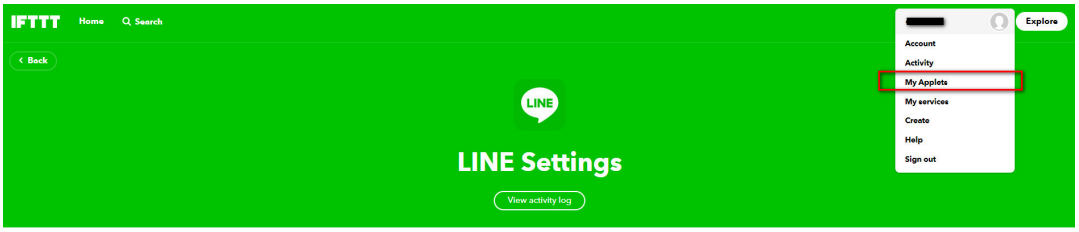
Step 4：连接完成会自动跳回 IFTTT 页面，单击右上方的【Settings】，可以看到服务状态 status 是 active 表示启用完成，如果对于连接有任何疑虑，也可以随时单击 Disconnect LINE 就能取消。



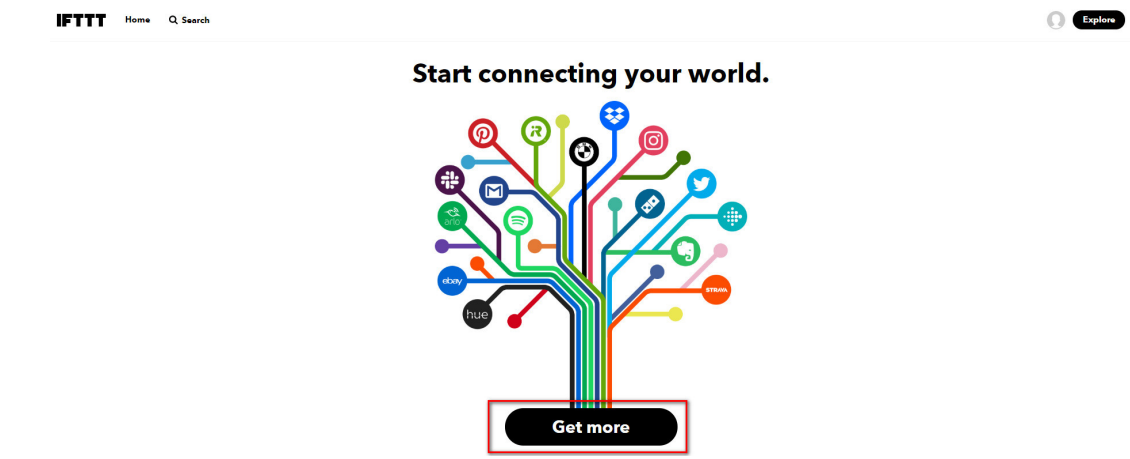


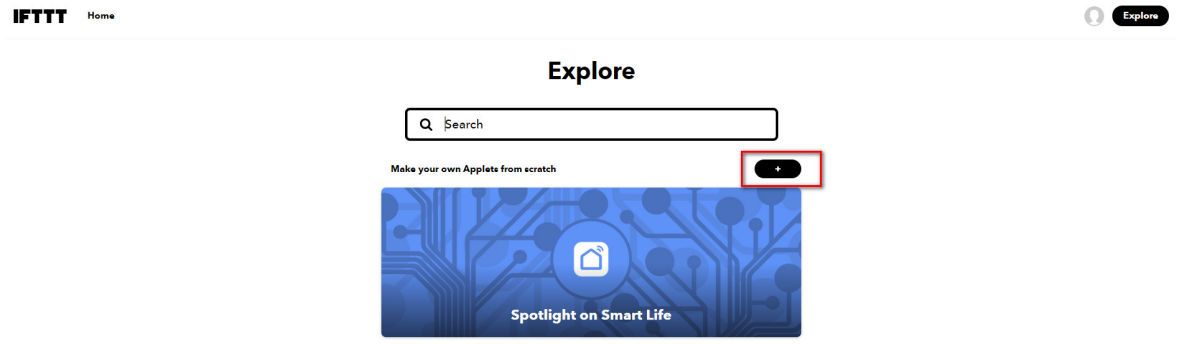


Step 5：建立好 LINE Notify 的连结后，单击右上【人像】，并单击【My Applets】，接着单击【Get More】，再单击 + ，以建立一个新的 Applet。





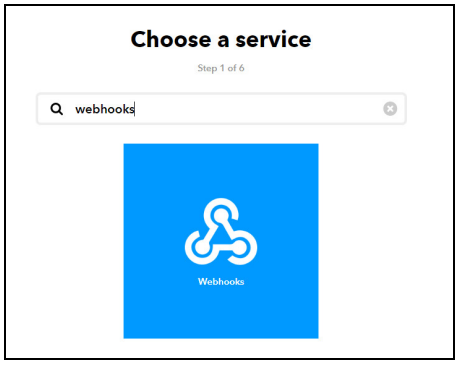


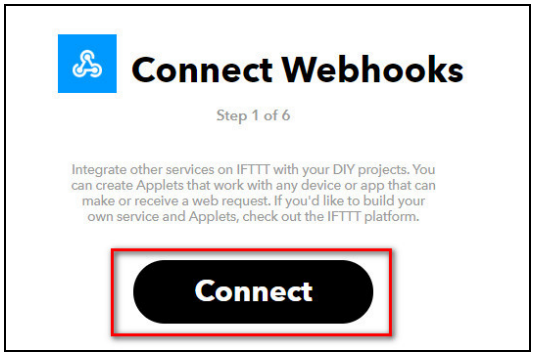


Step 6：单击后页面显示出现一段文字：If + This Then That，单击前方黑色的 + This。

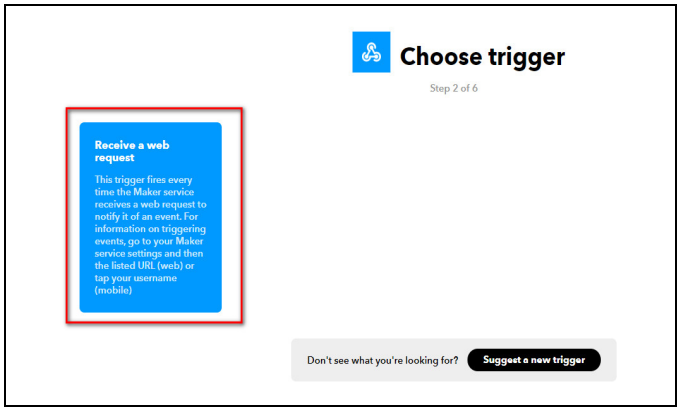


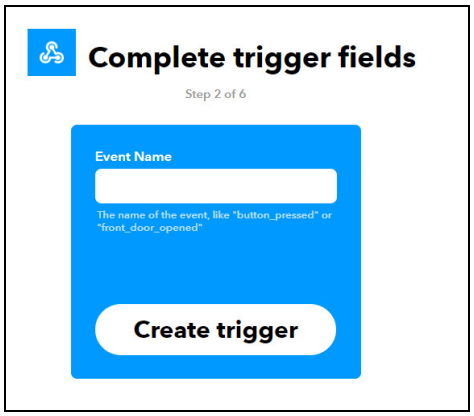
Step 7：单击后搜索并增加所需要的服务，在搜索的字段输入【webhooks】，搜索之后单击 Webhooks 图示，再单击【Connect】按钮。





Step 8：单击后，继续单击【Receive a web request】，会要求输入 Event Name（范例：输入名称为 line），输入完成按 Create trigger，就会回到前一个页面。

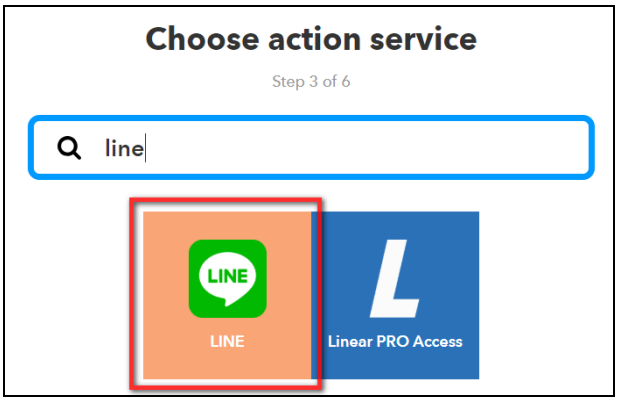




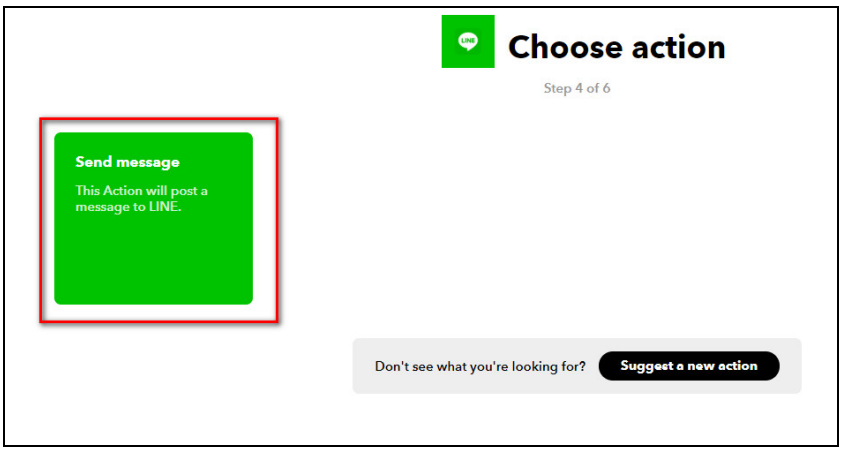
Step 9：回到前一个页面后，单击后方的 + That。



Step 10：在搜索字段输入【line】，搜索之后单击下方 line 的图示，添加 line 服务。



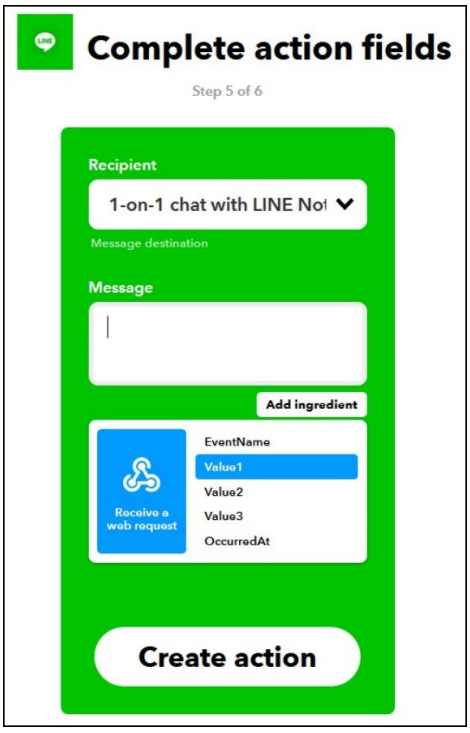
Step 11：单击 Send message。



Step 11-1：进入发送信息设定页面。

Step 11-2：单击 Recipient 的下拉选单，可以看到自己的 Line 群组清单，因此也可指定发送信息到所指定的群组，默认是一对一传送信息。

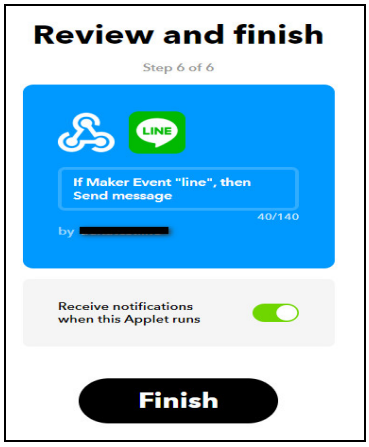
Step 11-3：Message 清空后，单击 Add ingredient 按钮，选择 Value1 即可。

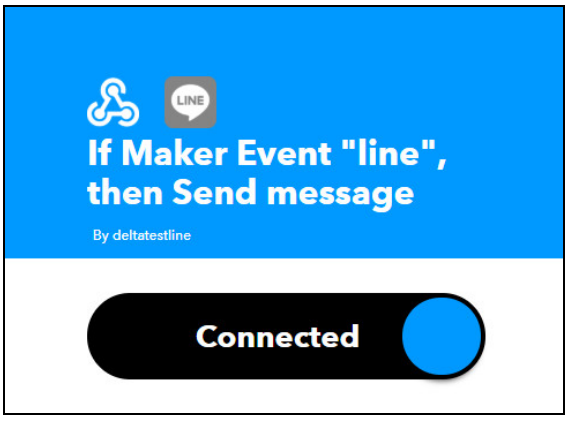


Step 11-4：单击【Create action】按钮进入下一步。

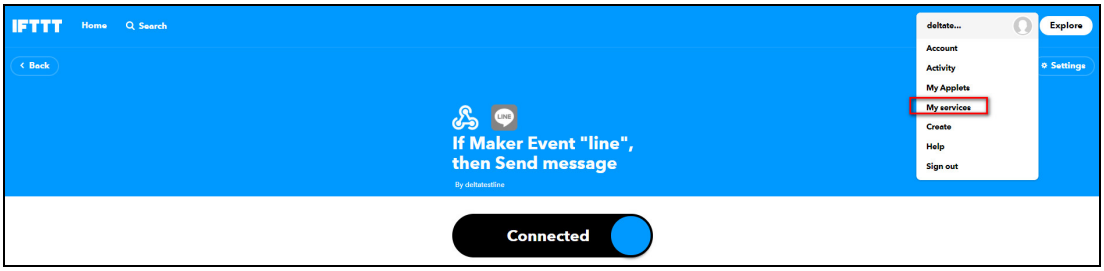


Step 11-5：进入【Review and finish】页面后，单击【Finish】即完成设定。



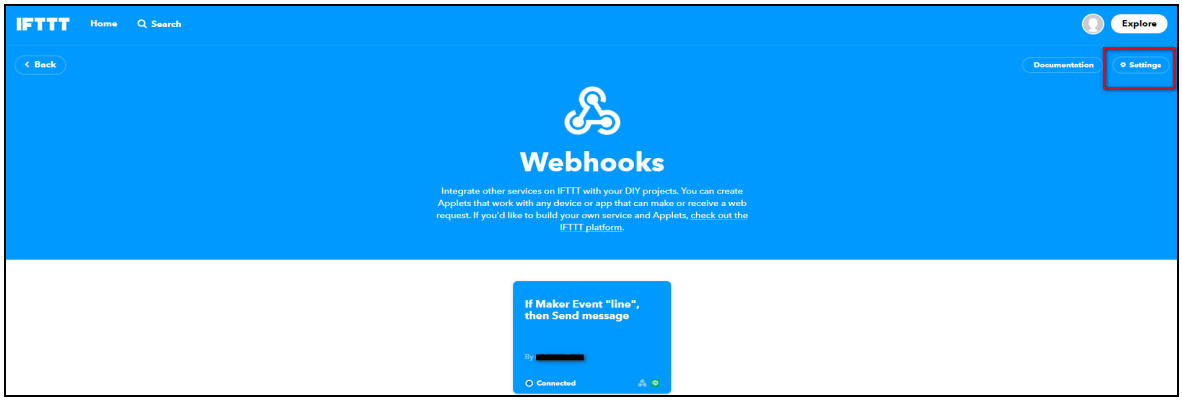


Step 12：右上单击【人像】，接着单击【My services】，于下一页单击 Webhooks。

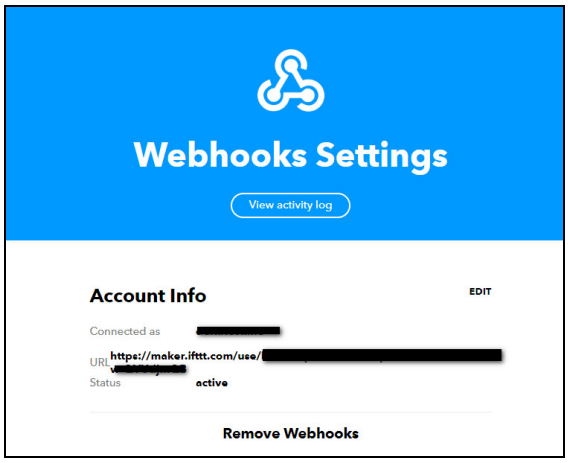




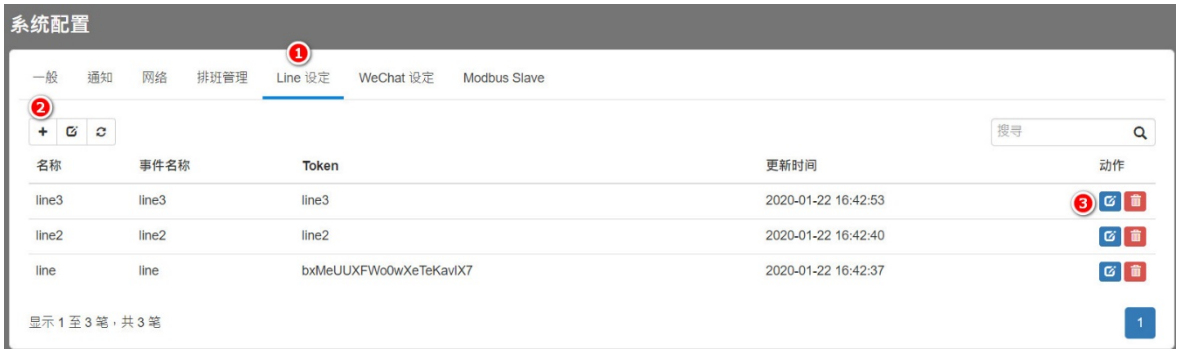
Step 13：进入 Webhooks 后单击右上方的【Settings】，在 URL 字段后方的乱码就是密钥。
（密钥十分重要，勿外流）



注意：若单击 Edit connection 按钮，密钥会重新产生，故需重新同步至系统！



Step 14：开启 DIALink 网页 > 左方选单【系统设定】> ❶【Line 设定】：显示“Line 设定列表”。在“Line 设定列表”页面，单击上方❷➕符号。



Step 15：按照以上步骤分别填入事件名称及 Webhooks Token，即可完成 Line 设定。

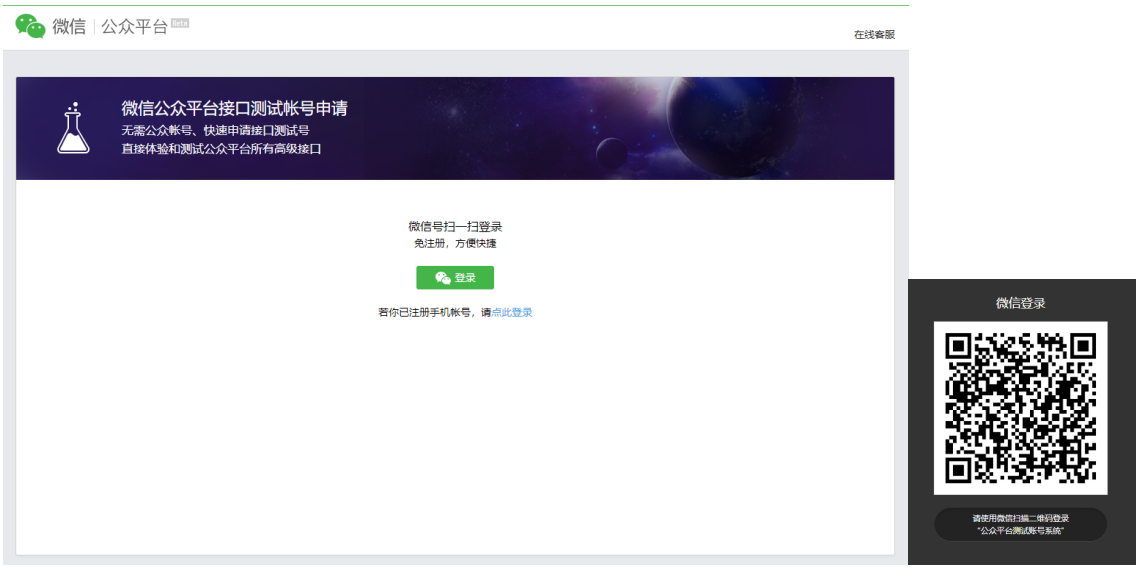
事件名称：必须等同 IFTTT 申请时填入的【Event Name】。Token：IFTTT Webhooks 密钥，密钥更新必须同步更新此字段内容。



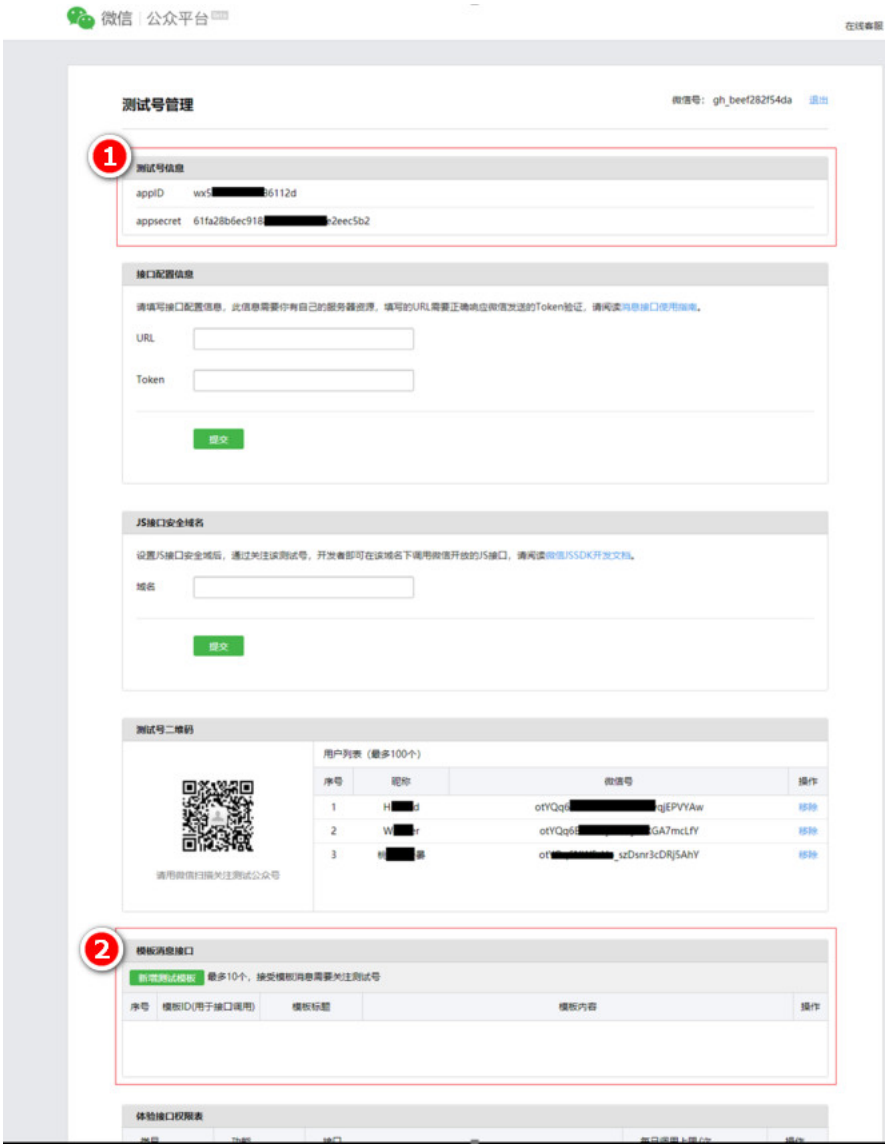
B

B.2 微信公众号申请教学

Step 1：进入微信公众平台接口测试帐号申请页面，单击登录按钮进行 QRCODE 扫描。



Step 2：登入后即可看到下方页面，体验接口下方信息由于页面长度有限，故将之移除，主要重点是红色框部分。



Step 3：如上图，第一个红框部份：测试号信息，appID 及 appsecret 是系统自动生成的，也是后续需要新增至 DIALink WeChat 信息的字段。

Step 4：如 Step 2 的图，第二个红框部份：模板消息接口，此部份需要于下方的 新增测试模板，

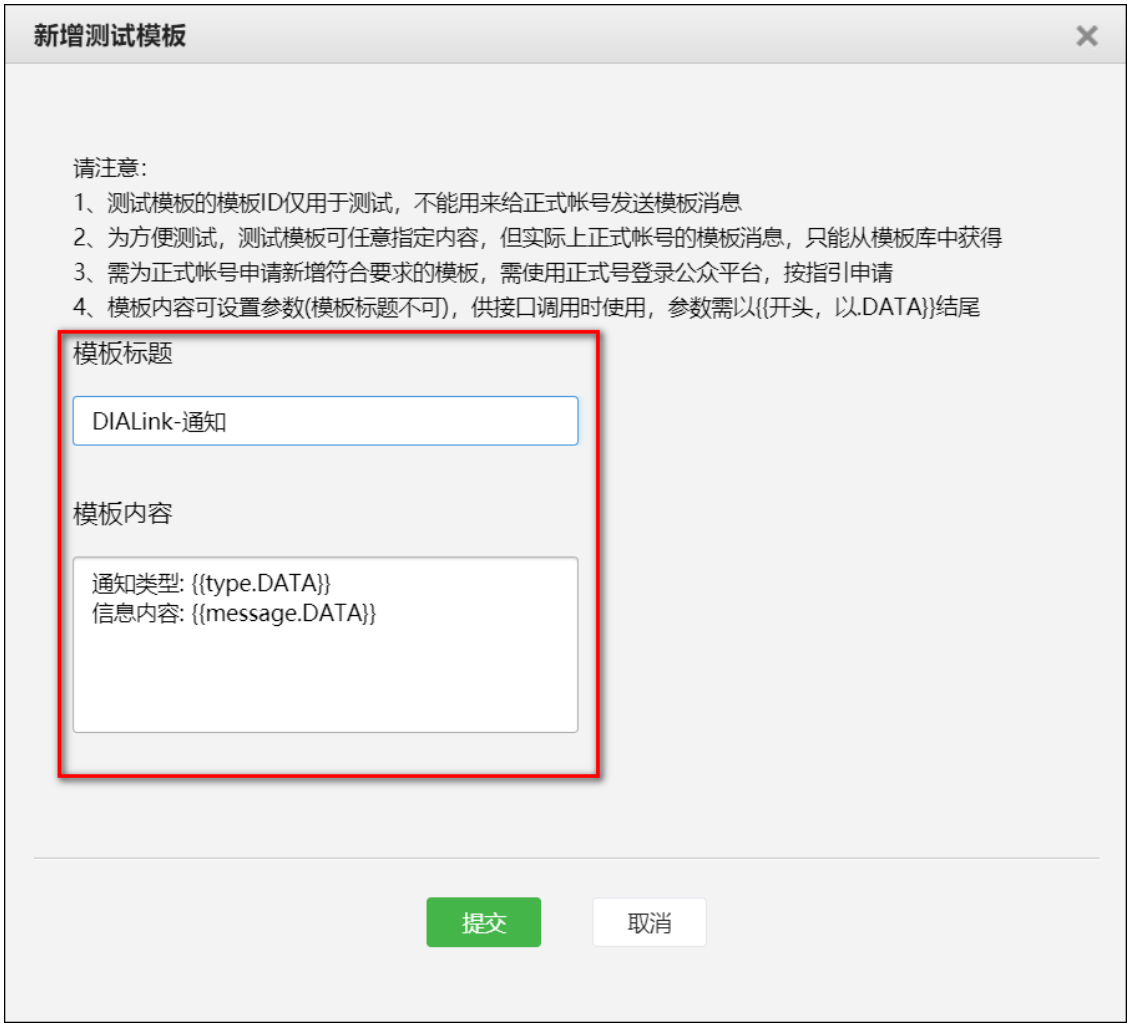
建立新的模版，并输入以下信息，其中模板内容需与下图模版内容一致，否则会导致微信信息无法送出。

模版标题：DIALink-通知

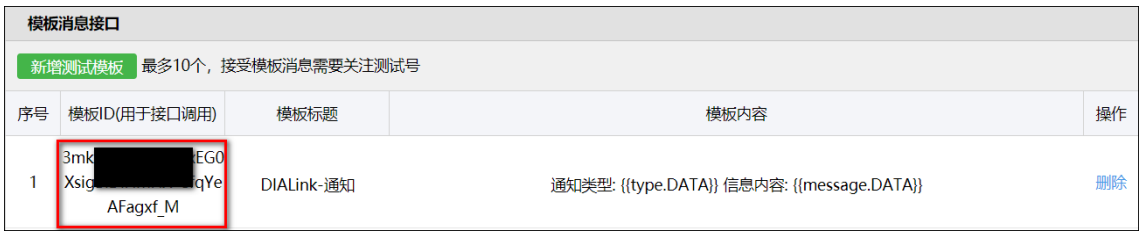
模版内容：

通知类型：{{type.DATA}}

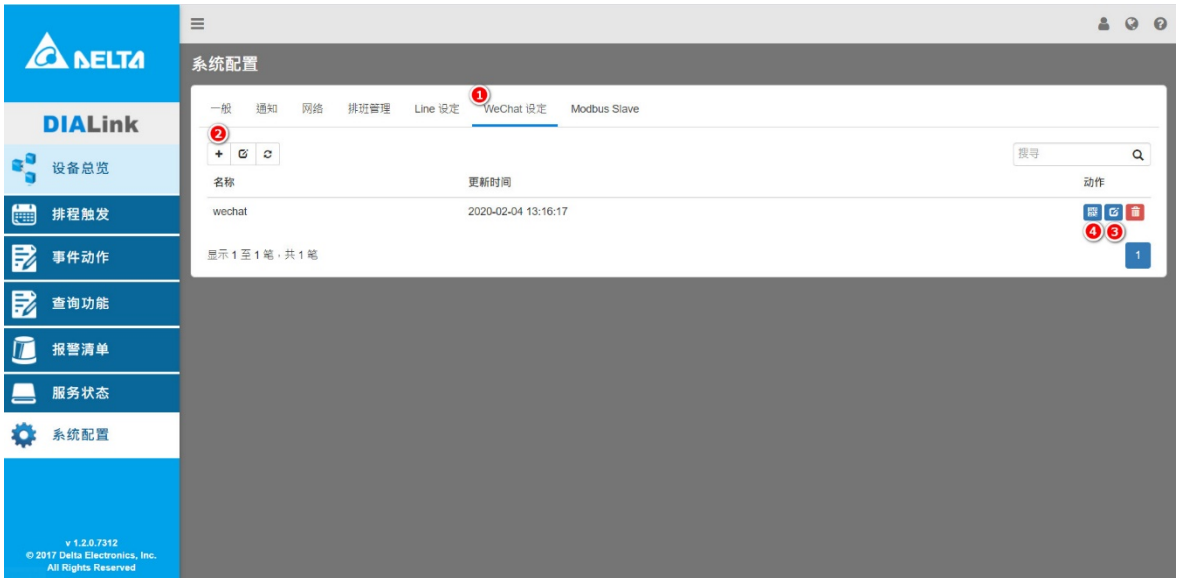
信息内容：{{message.DATA}}



Step 5：建立完毕后，模板消息接口会出现一笔数据，其中红框部份的模板 ID 是后续需要将之新增至 DIALink WeChat 信息的字段。



Step 6：开启 DIALink 网页 > 左方选单【系统设定】> ❶【WeChat 设定】：显示 “WeChat 设定列表”。于 “WeChat 设定列表” 页面，单击上方❷➕符号。



Step 7：将测试号信息（Step 3）中的 appID 及 appsecret 填入页面上的 Appid 及 Secret 字段，并且将模板消息接口（Step 5）中的模板 ID 贴入模板 ID 字段，输入完后，单击下方【保存变更】键。



C

附录C 设备点位导出导入功能

目录

C.1 概述 C-2
C.2 点位导出说明 C-2
C.3 点位导入说明 C-3

C.1 概述

点位导出导入功能，主要是提供用户导出设备所有点位数据，以 Excel 文件格式呈现。此文件为系统认可的模板。若用户需要针对设备所有点位进行一次性的多笔修改、新增，则需先下载此范本文件后，进行内容编辑，并于系统上进行导入动作。

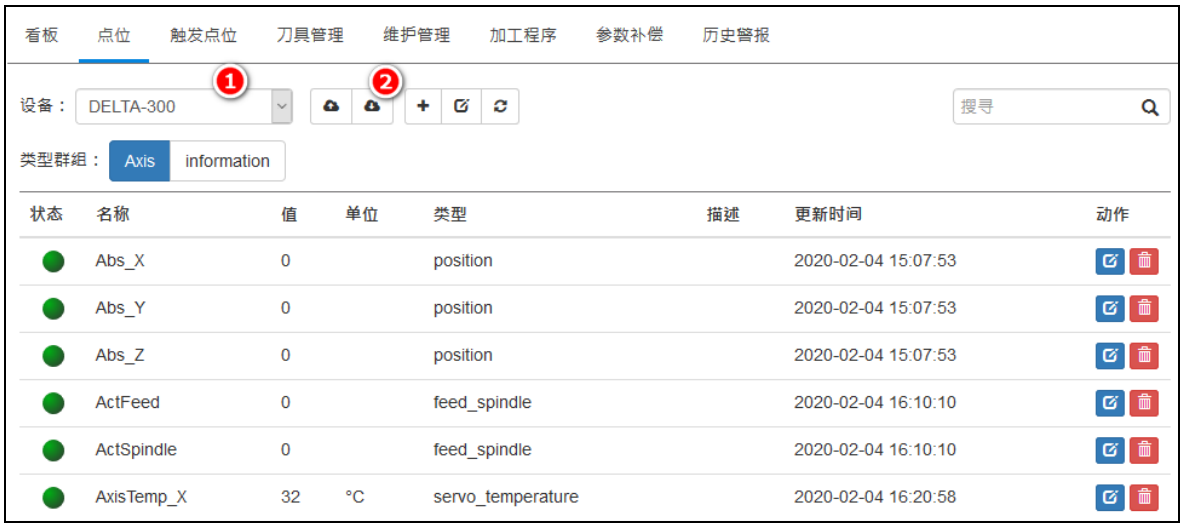
点位导出：各设备点位导出，以 Excel 文件呈现，为系统认可的模板。

点位导入：文件必须是认可的模板，导入的数据内容会被验证，若验证失败系统会弹出信息或回传导入的文件，提醒用户更正后再上传。

C.2 点位导出说明

导出设备所有点位数据，❶先筛选设备，❷单击  键进行导出。

单一设备导出最大点位数为 1 万笔。



导出的 Excel 文件为导入时使用的模板，主要分为四个工作表。

❶Tag List：点位数据

❷Type：点位类型参考。

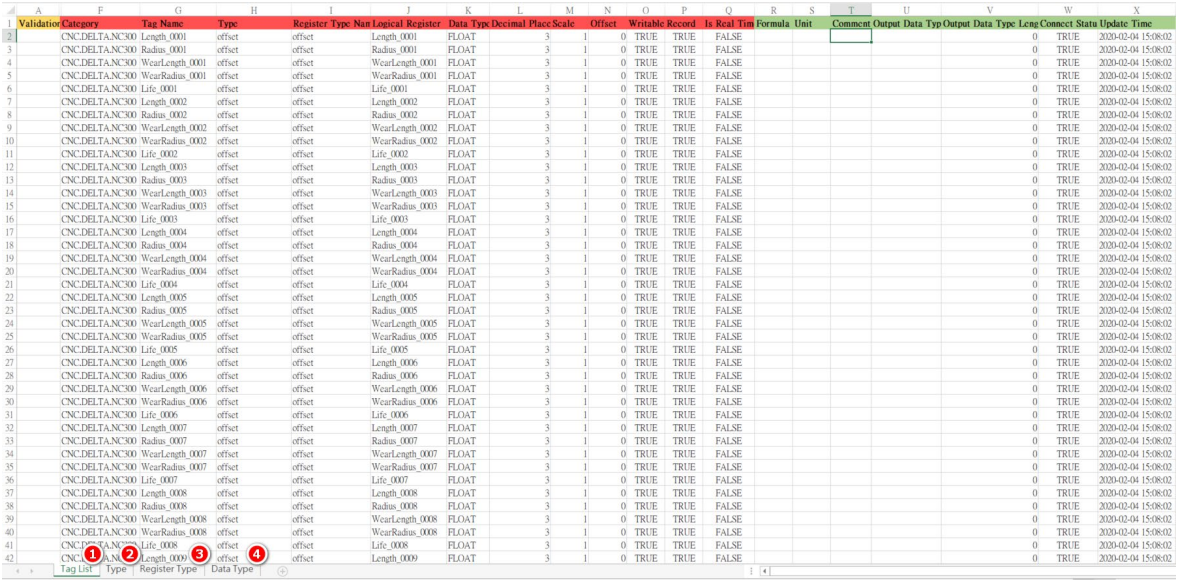
依据各设备类型（CNC，PLC，Modbus…）以及型号，点位类型也不相同。

❸Register Type：点位地址类型参考。

依据各设备类型（CNC，PLC，Modbus…）以及型号，点位地址类型也不相同。

❹Data Type：点位数据类型参考。

表示数据内容为字符串（STRING，….）或其它（INT，FLOAT…）



C.3 点位导入说明

导入设备所有点位数据，❶先筛选设备，❷单击  键选择导入的 Excel 文件，此文件必须是由系统导出的模板。**单一设备导入最大点位数为 1 万笔。**



模板操作及内容条件说明：

- 工作表名称【Tag List】、【Type】、【Register Type】及【Data Type】不可更改
- 工作表【Tag List】：

  字段抬头，**红色**：必填字段；**绿色**：非必填字段；黄色：各列必填字段验证结果。

  重要字段说明：

  - Validation：验证结果，失败字段显示【fail】；成功则显示【空白】。
  - Device Type：设备种类，可填入「**CNC**、**PLC…**」等等。

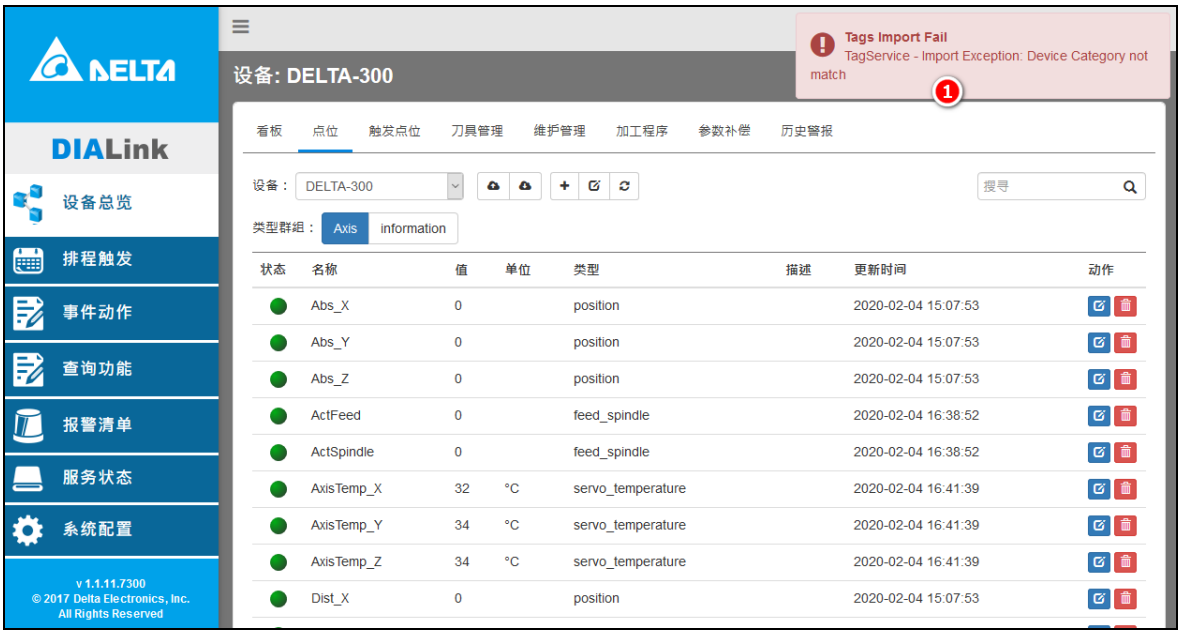
- Device Brand：根据设备种类不同，可填入的设备品牌也不同。
- Device Model：根据设备种类、品牌不同，可填入的设备型号也不同。
  （以上三项请参考 2.2.2 章节）
- Tag Name：**点位名称，不能重复。*依据点位名称判定此笔数据为新增或修改条件*。**
- Tag Type：点位种类，如下表所列：

Category	Tag Type
CNC Default Tag	work_rate, usedcount, status, time, feed_spindle, othercode, offset, usedtime, part, position
Edge Computing Tag	arithmetic, logic, counter, timer, move
Other Tag	Modbus, Parameter, Macro, OPC, PLC, OPCDA, OPCUA, PC, Robot

- Register Type Name：点位地址类型。依据各设备类型（CNC，PLC，MODBUS…）以及型号，点位地址类型也不相同。详细内容可参考工作表【Register Type】。
- Logical Register：点位地址，例如：D: 0、 X:2、 16 等等。
- Data Type：点位数据类型。例如：STRING、FLOAT、INT、INT (LITTLE-ENDIAN)….。
- Decimal Place：点位数据类型为 FLOAT。数值小数位数，范围：0～8。默认值，请带入 3。例如：点位数值为 123.456，此字段值为 2，运算后数值为 123.45。
- Scale：点位数值放大或缩小倍数。默认值，请带入 1。例如：点位数值为 123，此字段值为 10，运算后数值为 1230。若点位数值为 567，此字段值为 0.01，且点位数据类型为 FLOAT，运算后数值为 5.67。若点位数值为 567，此字段值为 0.01，且点位数据类型为整数类，运算后数值为 5。
- Offset：点位数据位移（数值加/减运算）。默认值，请带入 0，例如：点位数值为 123，此字段值为 20，运算后数值为 143。
- Record：点位数据是否记录于数据库，TRUE：记录；FALSE：不记录。
- Writable：点位是否可回写到设备端，TRUE：可写；FALSE：只读。
- Is Real Time：针对 CNC 点位可分成是否为实时更新数值，例如刀具补偿设定为 False，其他设备种类点位则一律设定为 True。
- Cloud Transmission：是否传送设备点位值数据至云端平台，例如：Azure 及 AWS。可设定值为 0：None（不传送），1：Interval（固定频率传送），2：Change（点位值更新才传送）。
- First Number：针对 Counter 跟 Timer 的虚拟运算点位初始值。
- Increment Number：针对 Counter 的虚拟运算点位递增值。
- Formula：虚拟运算点位运算/逻辑公式。
- Unit：点位单位显示。
- Comment：点位描述。

- Output Data Type：目前尚未使用，可不用填写。
- Output Data Type Length：若点位数据类型为 BIT 时，此字段会取出该设备指定地址数值的第几个 BIT（数值范围 0-15）；若点位数据类型为 ASCII / ASCII(HW/LW)时，则表示此文字长度为多少，这里的 Length 为 1 表示要抓两个英文字或两个 ASCII。例如：点位数据类型为 BIT，此字段为 3。此点位从设备读到数值为 140，则此点位数值运算后为 1(True)。
- Connect Status：可设定当设备联机时，是否读取此点位。
- Update Time：更新时间。

- 若导入的文件内容错误，系统会以❶弹出信息或回传问题文件提醒用户修正后再重新执行导入动作。



MEMO

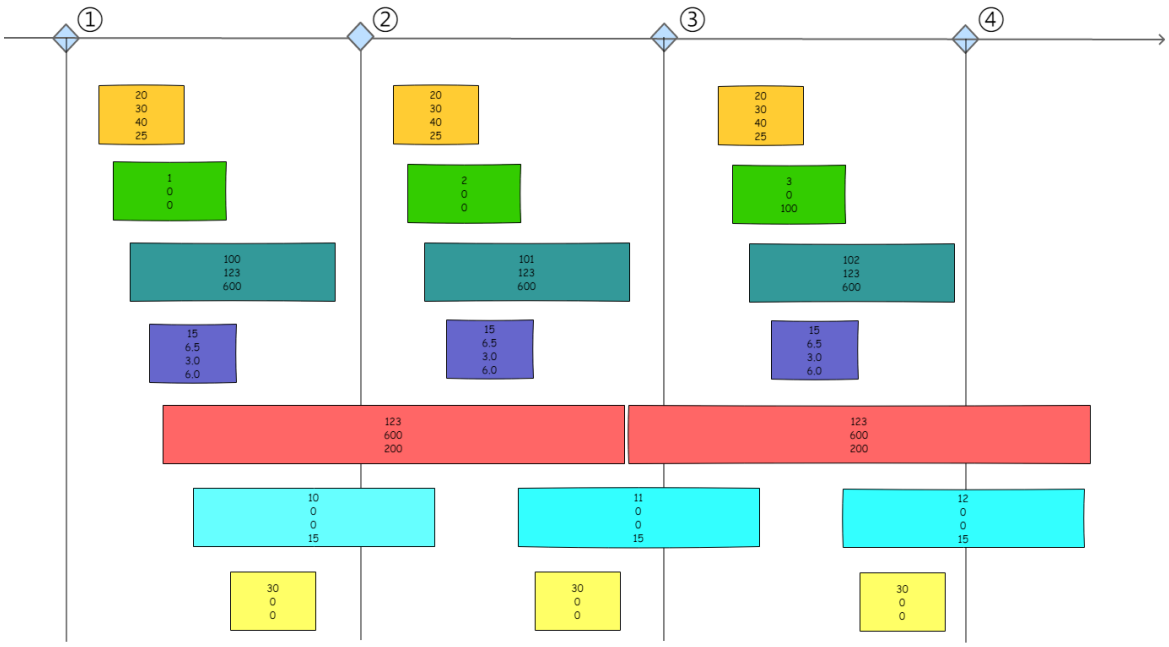
D

附录D 点位计算功能数值有效时间设定

目录

D.1 概述 D-2

D.1 概述



从网页上可以设定系统搜集数据的间隔时间，叫做「扫描区间」，可在节点的属性设定，然而每个设备能响应数据的时间长短不一，有的设备读取时间会超过这个扫描区间，这时就要考虑到，虚拟点位的结果要拿前次时间的数据来计算，或是直接将结果存为 Null。

以上图来看，每个色块代表一个设备，上面的数字则是要抓取几个点位的数值，由于程序执行后并不会主动抓取数据，而是在收到 MQTT 传来的设备与点位信息之后才会去读取数据，所以并不会同一时间同时去读取设备，虽然每个设备的开始读取的时间并不一定。但是第一次送达的设备信息会包含系统所有设备，理论上系统刚开始每个设备读取时间的差异不大。然而在执行之后变更或新增设备的点位就会跟其它点设备的读取时间有落差了。

而每个设备会在读取完所有的点位之后才会把该设备所有点位的数值一次送出(送给 MQTT 再让需要的模块订阅取用)，并且等待到「扫描区间」的时间到了，才会再次去读取数据，所以读取状况就如同上图了。也就是说，在同一设备上所有点位都会在同一时间点完成读取，不会有某些点位先读取完成，有些点位后读取完成的时间差发生。

由于读取设备是被动触发，无法统一所有读取设备的时间，很难达到完美切分出 DataCollector 去读取设备的时间，依照上图，假设有个虚拟点位的公式是所有颜色的设备的第一个点位相加，那么在②的时间点，红色的设备的数值就会缺值，一直到③的时间点才能读取到，所以若有设备的读取时间是很慢的，那将可能导致所有引用到该设备的虚拟点位都会变成 NULL，若无法克服该设备的效率，则只能通过设定将「扫描区间」的时间拉长。

系统将提供让用户自行决定「使用前值」、「显示断线」两种模式，在公式内若有点位数值无法实时取得时，要将公式断定为 NULL 或是拿一个数值来计算，针对不同的客户需求，都能够满足。在网页上新增一个接口可以让用户选择这两种模式，而「显示断线」会保存设定为：「asnull」，「使用前值」则为：「lastvalue」。

MEMO

附录E OPC UA Server 设定

目录

E.1 需求及环境架构说明 E-2
E.2 OPC UA Server 服务及设定 E-2
E.2.1 OPC UA Server 服务状态 E-2
E.2.2 系统设定 E-3
E.2.3 新增设备及点位 E-4
E.3 UAExpert 安装 E-8

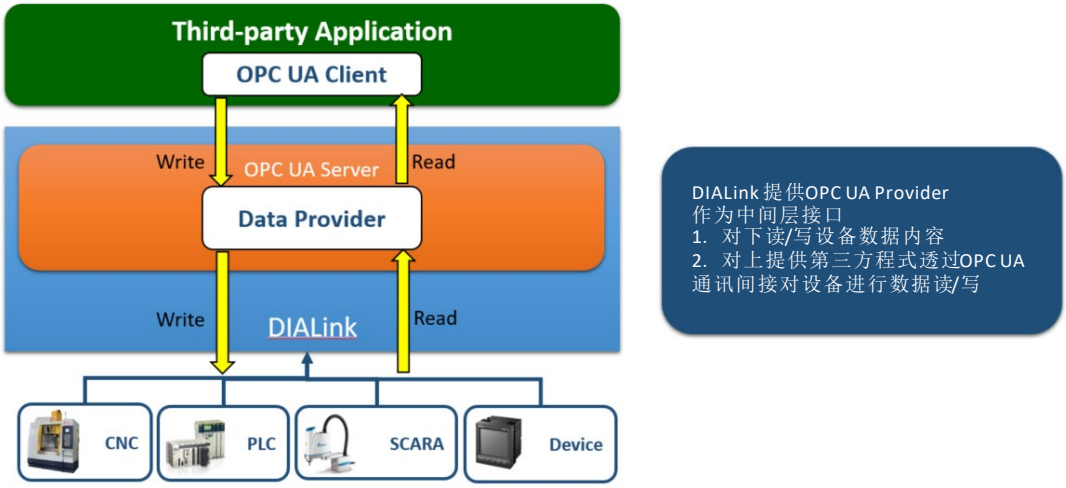
E.1 需求及环境架构说明

DIALink 支持 OPC UA 通讯，以 OPC UA Provider 为中间层接口（下图）。主要作为：

- 对下读/写设备数据内容
- 对上提供第三方程序通过 OPC UA 通讯间接对设备进行数据读/写

以上数据更新记录也会呈现在 DIALink UI。

【注】仅供第三方程序读 / 写功能设备数据。

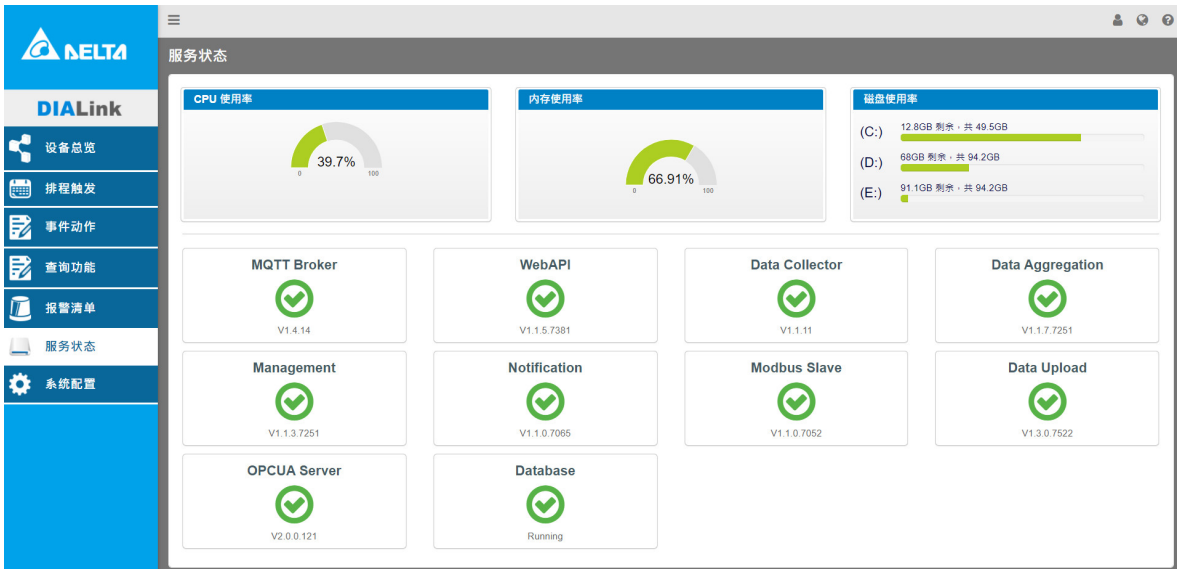


E.2 OPC UA Server 服务及设定

E.2.1 OPC UA Server 服务状态

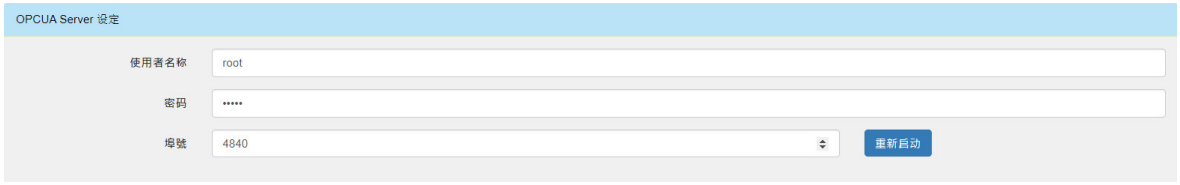
DIALink「服务状态」页上可看到一个「OPCUA Server」，这个模块负责提供第三方系统(ex: MES, SCADA) 通过 OPC UA 通讯下命令对设备请求数据写入、读取收集上来的数据。

系统正常启动时，所有服务会显示绿色勾勾，状态图标下方显示版本编号；服务未启动会显示红色叉叉，状态图标下方显示 Not Started。



E.2.2　系统设定

至「系统设定」,「一般设定」页面，用户可根据自身需求设定相关参数。提供登入名称、密码及通讯端口的变更。系统默认用户名称 / 密码：root / admin，通讯端口：4840



OPC UA Server 规格限制：

节点(Node)新增 / 删除或属性内容只要更新，须重启 OPC UA Server，Client 端才能重新浏览变更后的内容。因此用户在 DIALink 上新增、修改及删除设备或点位内容 (属性) 后，确认要提供第三方程序进行 OPC UA 通讯串接时，必须手动单击「重新启动」执行 OPC UA Server 重启，若是点位数值更改，OPC UA 则无须重启。

OPC UA Server 联机安全配置：

User Verified	Support
Anonymous	×
Account/PW	✓
Certification	×



Security Policy		Support
None	None	✓
Basic128Rsa15	Sign	✓
	Sign & Encrypt	
Basic256	Sign	✓
	Sign & Encrypt	
Basic256Sha256	Sign	✓
	Sign & Encrypt	

- 通讯格式：仅支持 UA TCP (opc.tcp://)，不支持 UA HTTP/HTTPS

- 用户验证：用来确认客户端登入时对数据存取的权限设定
- 安全策略：将信息以指定算法进行加密
- 目前仅支持 读 / 写功能
- Client 端同时最大联机数：5
- 同时监看 1000 点 (Tag/Node)，系统 CPU 使用率约占 15~20 %

E.2.3 新增设备及点位

- 新增设备

登入 DIALink 后，依序单击❶左方【设备总览】 > ❷【新增一台设备】，若已新建设备，单击右上方➕符号。



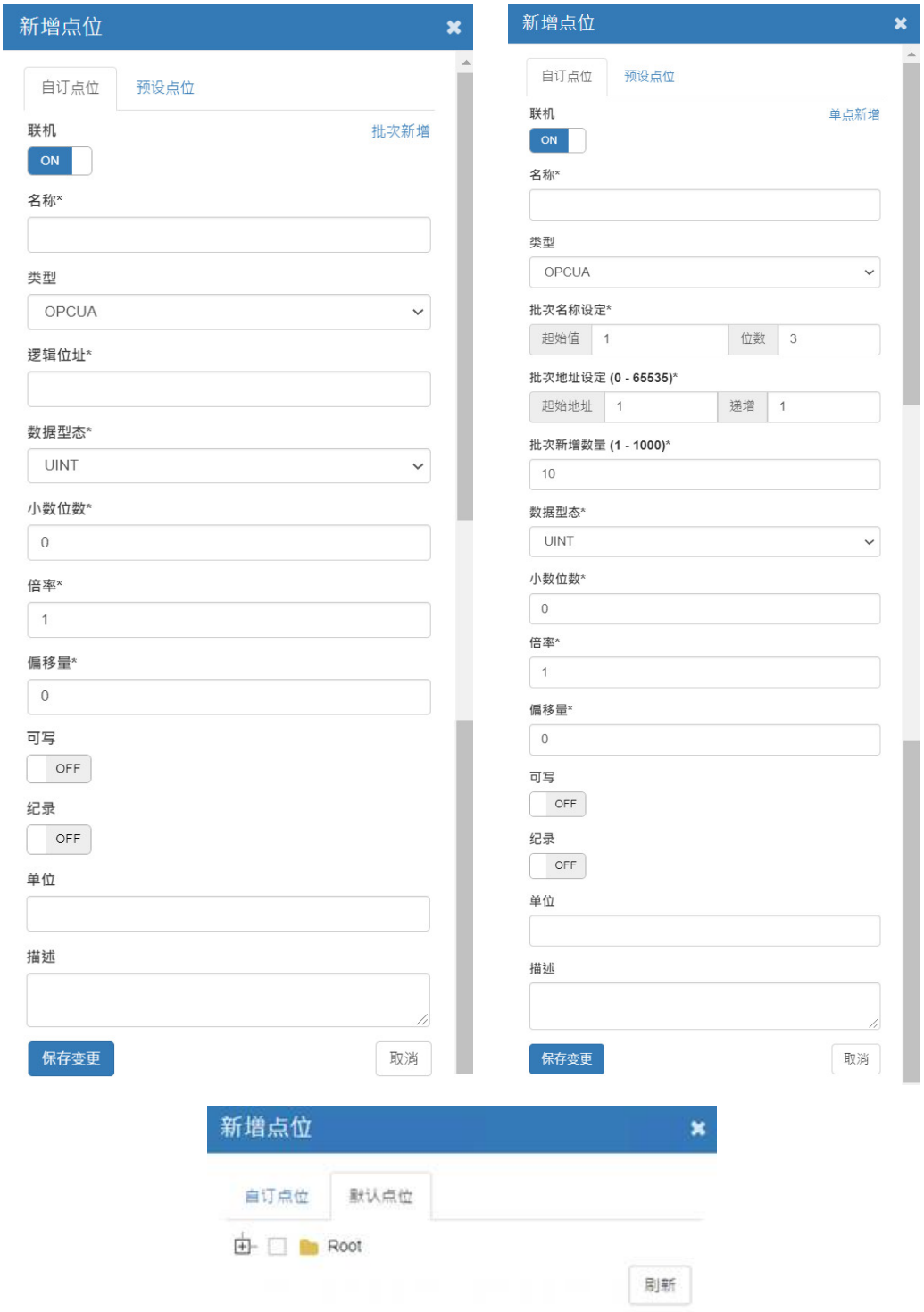
- 新增点位

选择❶左方【设备总览】后，单击❷设备节点内已新增的设备以进入点位设定页面。



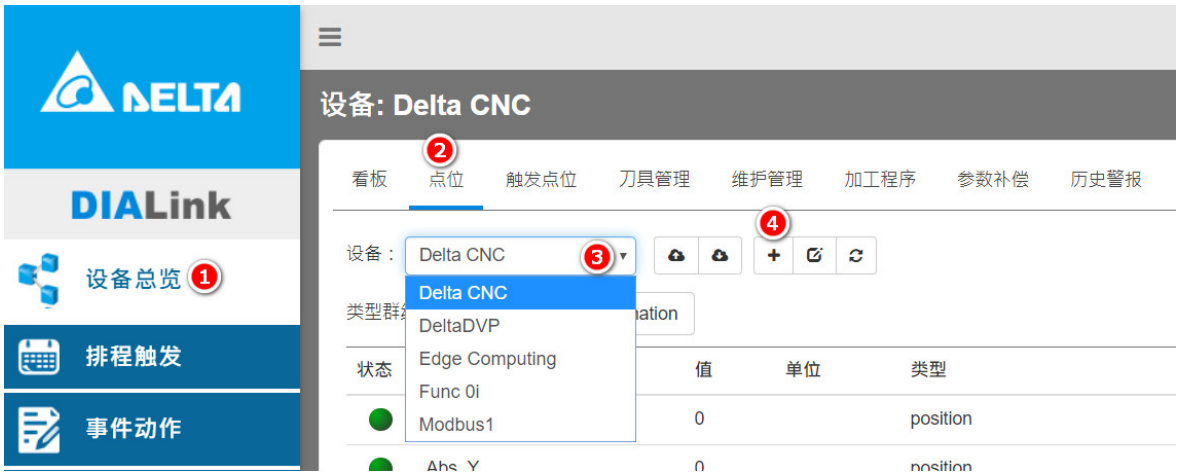
❷于【点位】页面> ❸切换设备> ❹单击  进行新增点位，页面右方显示【新增点位】相关参数设定，默认为单点新增，另一为批次新增，装置字段选择已建立的设备名称，根据不同设备类型，系统会带出相对应的参数，用户根据需求自行选择 / 输入设备相关参数。





星号（ * ）为必填项目，输入完毕单击下方【保存变更】键；若输入值不符合系统规则，表格边框会变为红线，并提示错误信息。

点位联机可随时设定关闭或开启，相关参数新增或编辑，皆须单击参数页面下方的【保存变更】键。



点位新增完毕后，单击【设备区块】中的【点位】分页，可于第一栏字段查看装置联机状态，第二栏字段为点位名称，第三栏字段为实时数值，如下图 CNC 控制器为例。



若要再次编辑点位参数，可于【点位】分页的动作字段下，单击 键进行编辑，完成编辑后单击下方【保存变更】即可；欲删除单一点位，可单击动作栏字段的 键进行删除。



若要删除多笔点位，❶先筛选设备，❷单击 键进行批次编辑，❸再勾选左方字段方框选择欲删除项目，❹单击 删除 键进行批次删除。



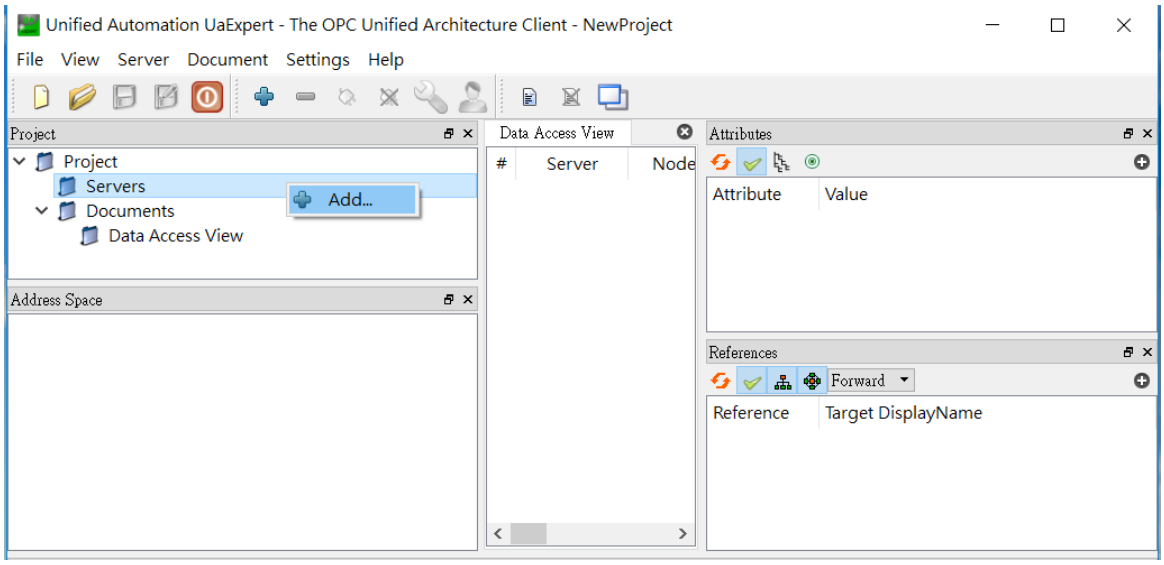
E.3 UAExpert 安装

UAExpert 是一款第三方 OPC UA Client 软件。

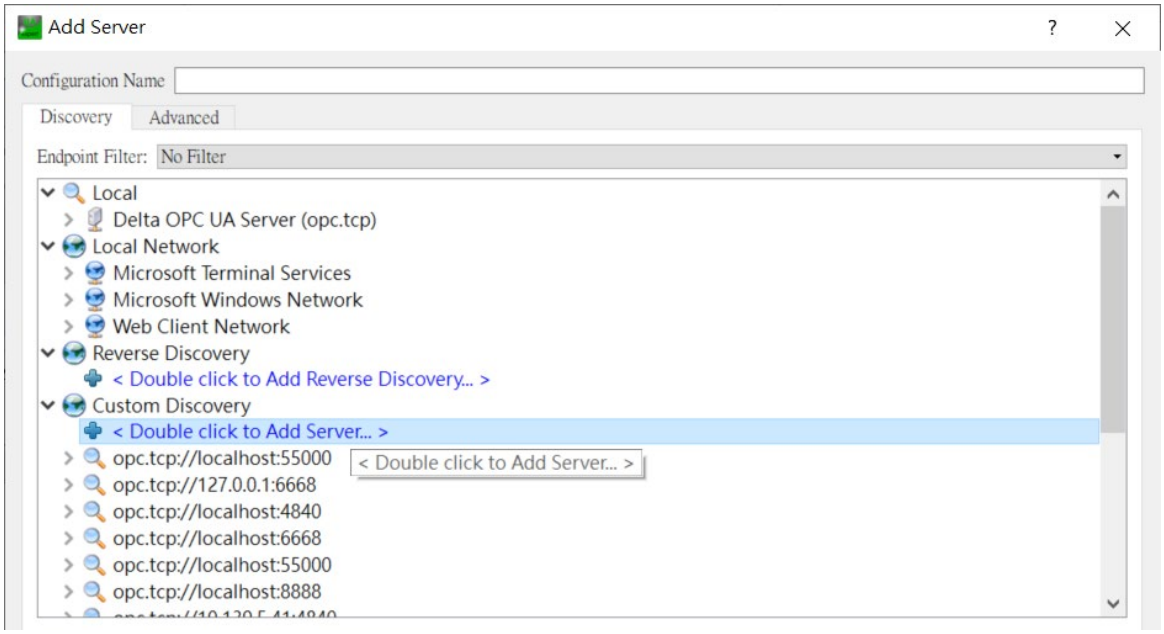
- 目前安装版本为 1.5.1
- 安装执行环境为 Windows 10 系统，Intel Core i3 双核 1.90GHz，4GB RAM
- 官网下载网址：https://www.unified-automation.com/downloads/opc-ua-clients.html
- 设定 Server 联机信息，通过用户名称 / 密码登入，系统默认为 root / admin。

UAExpert操作方式如下：

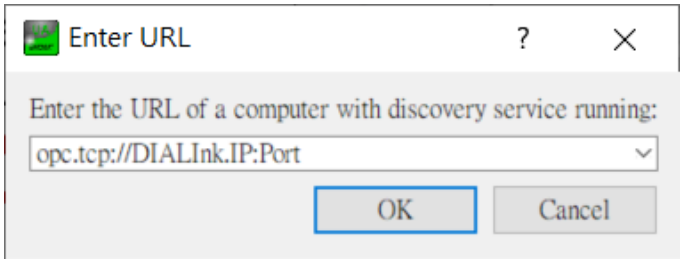
❶Project > Servers > 选择 Add...



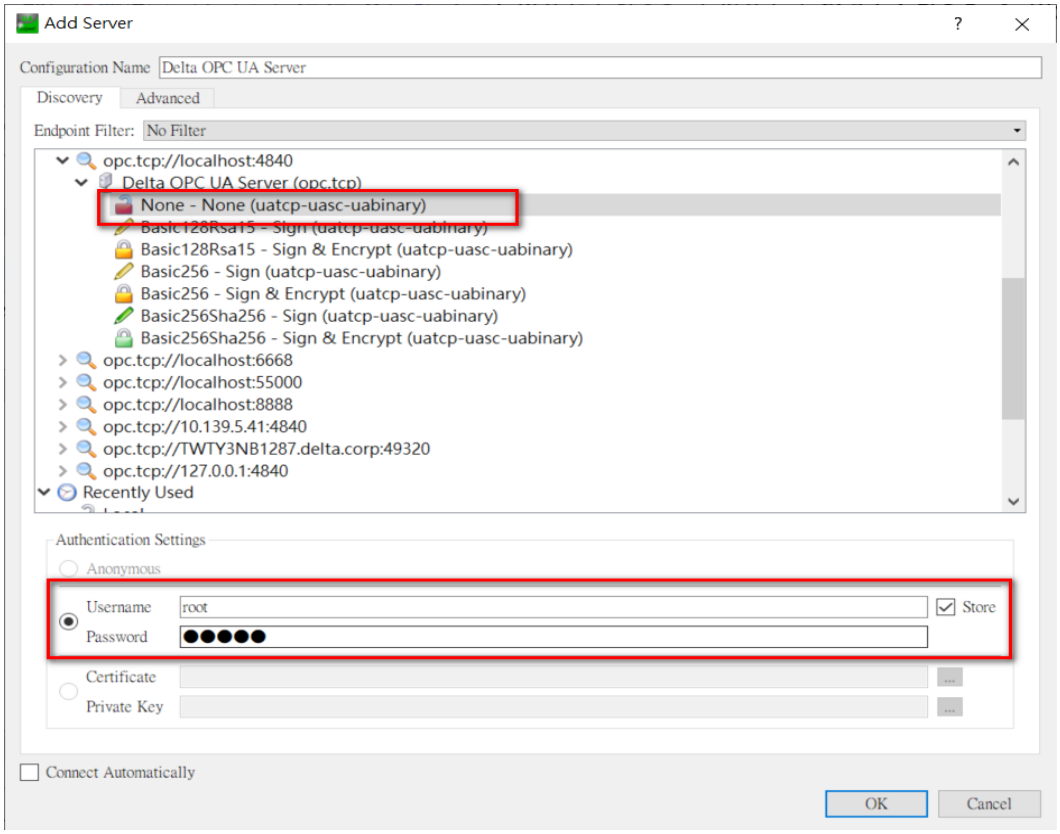
❷双击鼠标左键来加入 server



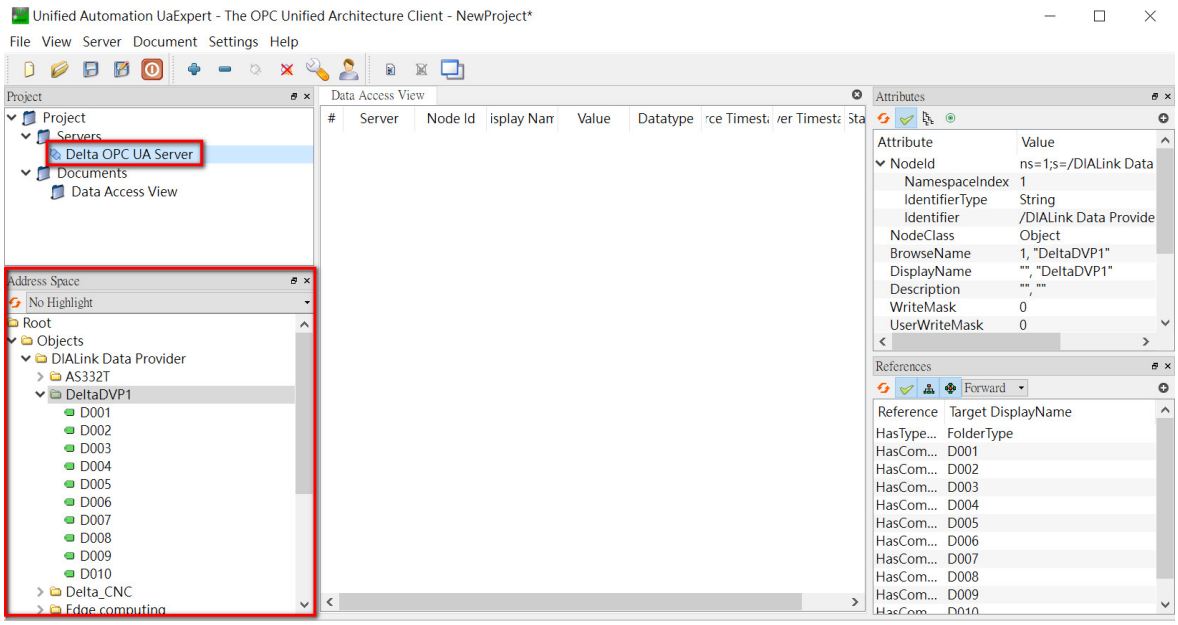
❸输入 URL: opc.tcp://DIALink.IP:Port



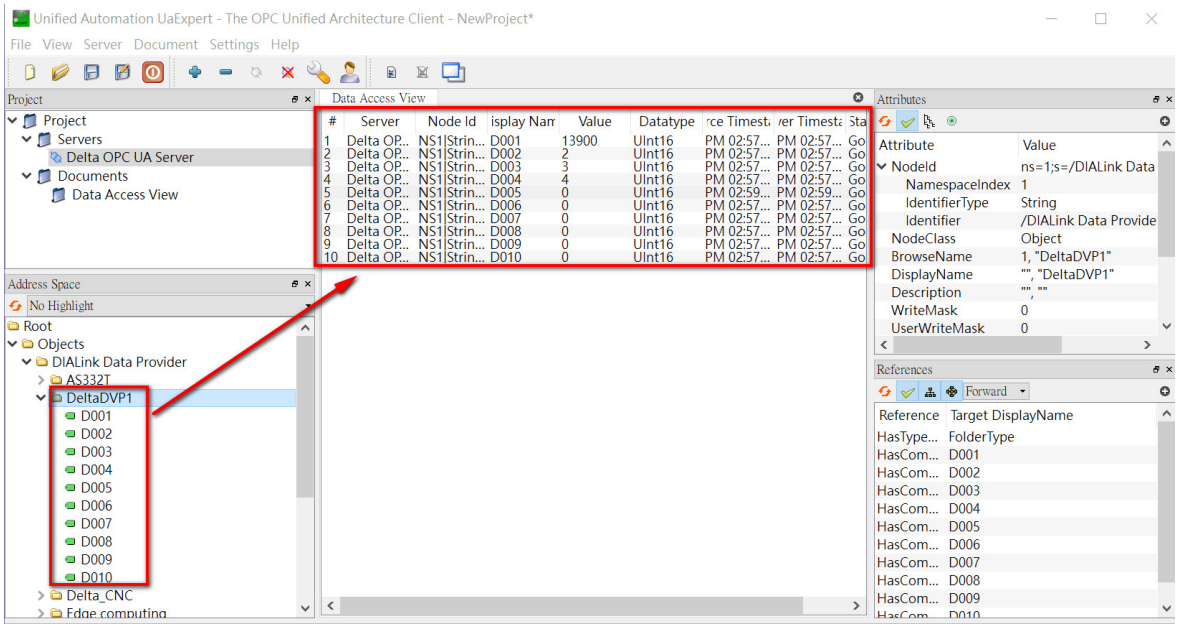
❹选择登入方式：默认请选择不加密，用户名称及密码登入



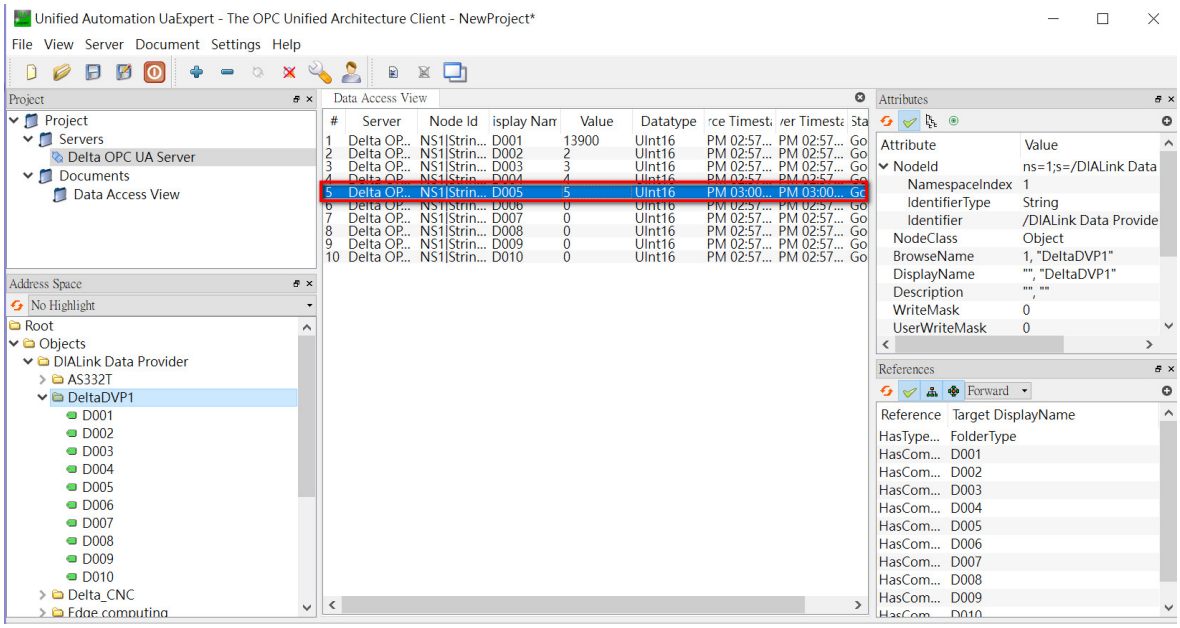
❺联机后会显示OPC UA Server 及Node内容



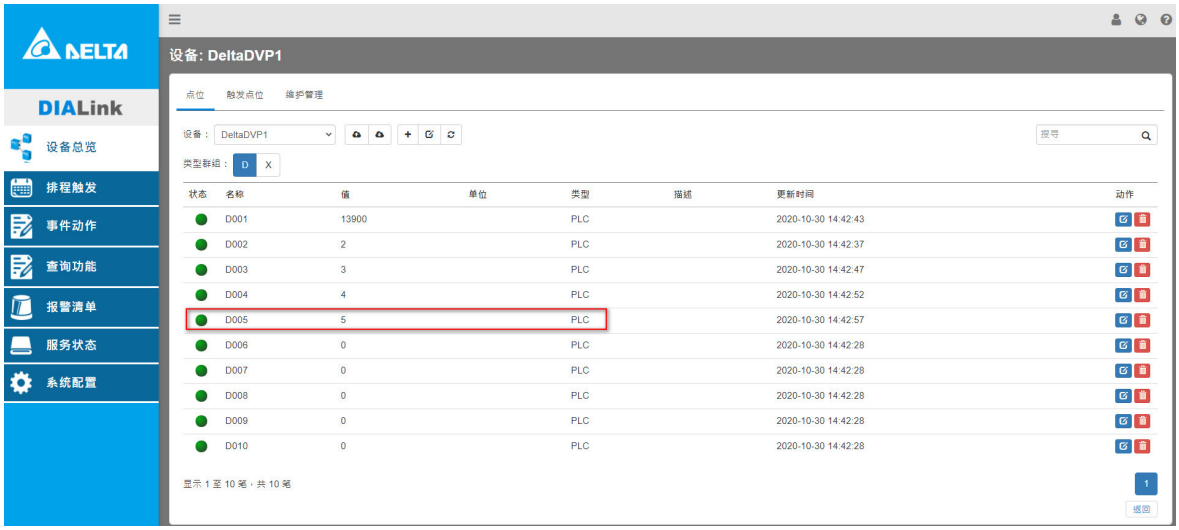
❻拖曳Node到中间窗口Data Access View可监看各Node数值内容变化



❼可修改Node数值内容（点位属性必须是可写入）



❽于DIALink网页，确认设备的点位数值是否有更新


DELTA
DIALink
设备总览
排程触发
事件动作
查询功能
报警清单
服务状态
系统配置
设备: DeltaDVP1
点位
触发点位
维护管理
设备：
DeltaDVP1
类型群组：
D
X
状态
名称
值
单位
类型
描述
更新时间
动作
D001 13900 PLC 2020-10-30 14:42:43
D002 2 PLC 2020-10-30 14:42:37
D003 3 PLC 2020-10-30 14:42:47
D004 4 PLC 2020-10-30 14:42:52
D005 5 PLC 2020-10-30 14:42:57
D006 0 PLC 2020-10-30 14:42:28
D007 0 PLC 2020-10-30 14:42:28
D008 0 PLC 2020-10-30 14:42:28
D009 0 PLC 2020-10-30 14:42:28
D010 0 PLC 2020-10-30 14:42:28
显示 1 至 10 笔，共 10 笔
1
返回

附录F 合并 Dongle Key 操作指南

目录

F.1 需求说明 F-2
F.2 合并工具 (DIALinkCombinDongleKeysForm) 操作说明..... F-2
F.2.1 扫描目前计算机 Dongle Key 信息 F-2
F.2.2 合并 Dongle Key F-4

F

F.1 需求说明

客户根据需求购买 DIALink，默认会有一组 Dongle Key（主 Key）开通可使用的设备（Ex: CNC, PLC, …）数目及基本功能。当未来需要扩展设备数目或新增功能模块时，则需另外购入相对应的升级 Key。

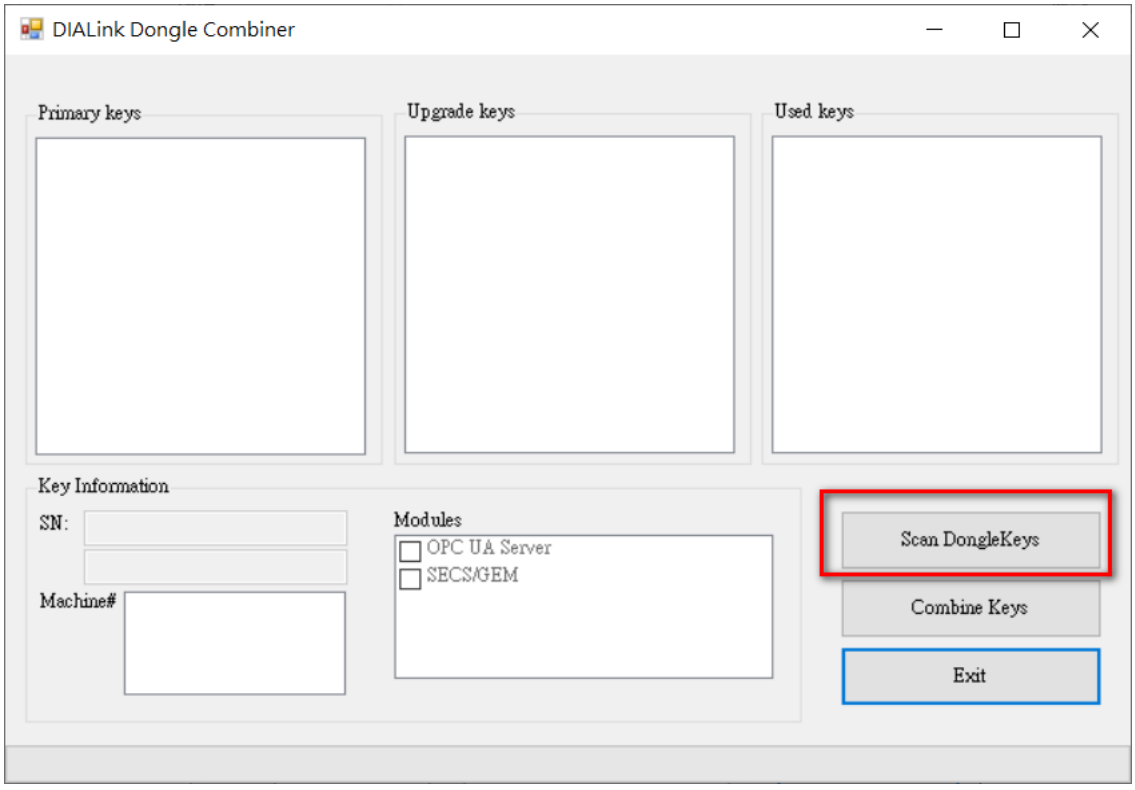
客户端通过工具（DIALinkCombinDongleKeysForm.exe）程序执行主 Key 与升级 Key 合并，合并后的主 key 则包含升级后的设备数或功能模块，升级 Key 则标记为使用过的 Key，无法再使用。

F.2 合并工具（DIALinkCombinDongleKeysForm）操作说明

F.2.1 扫描目前计算机 Dongle Key 信息

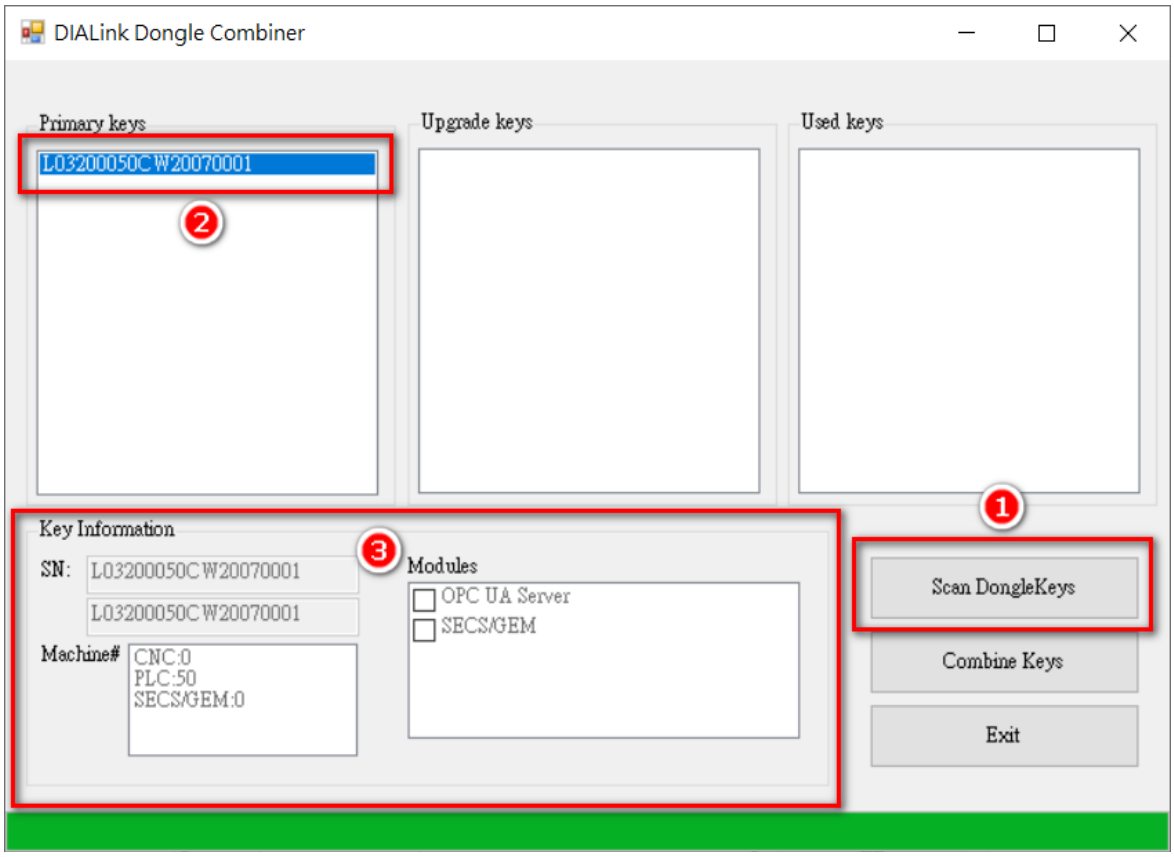
Step 1：执行工具程序并扫描

执行工具程序后，按下【Scan DongleKeys】，若扫描到合法的 Dongle 则会根据内容显示于对应的窗口内（Primary keys / Upgrade keys / Used keys）。



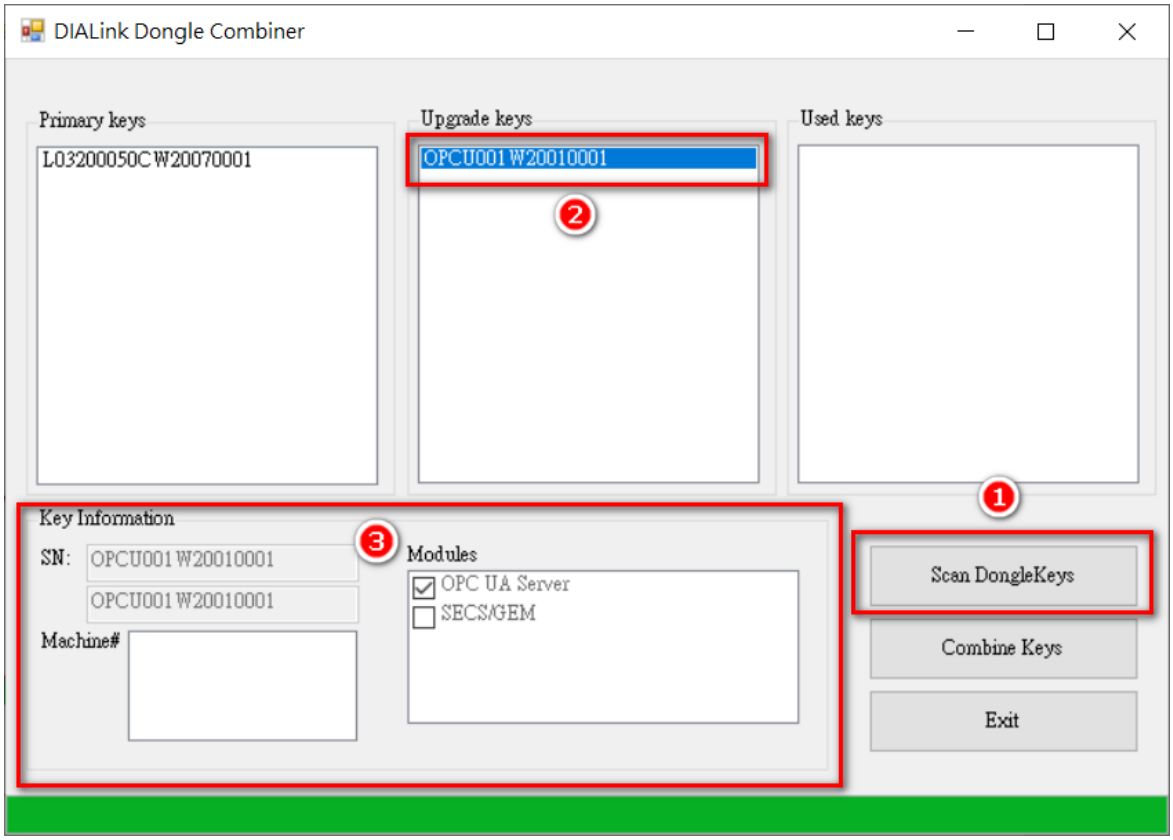
Step 2：插入主 Key

购入 DIALink 时附带的 Dongle Key。❶按下【Scan DongleKeys】后将显示于❷窗口【Primary Keys】内，单击后于❸【Key Information】区域内显示相对应设备数及功能模块内容。



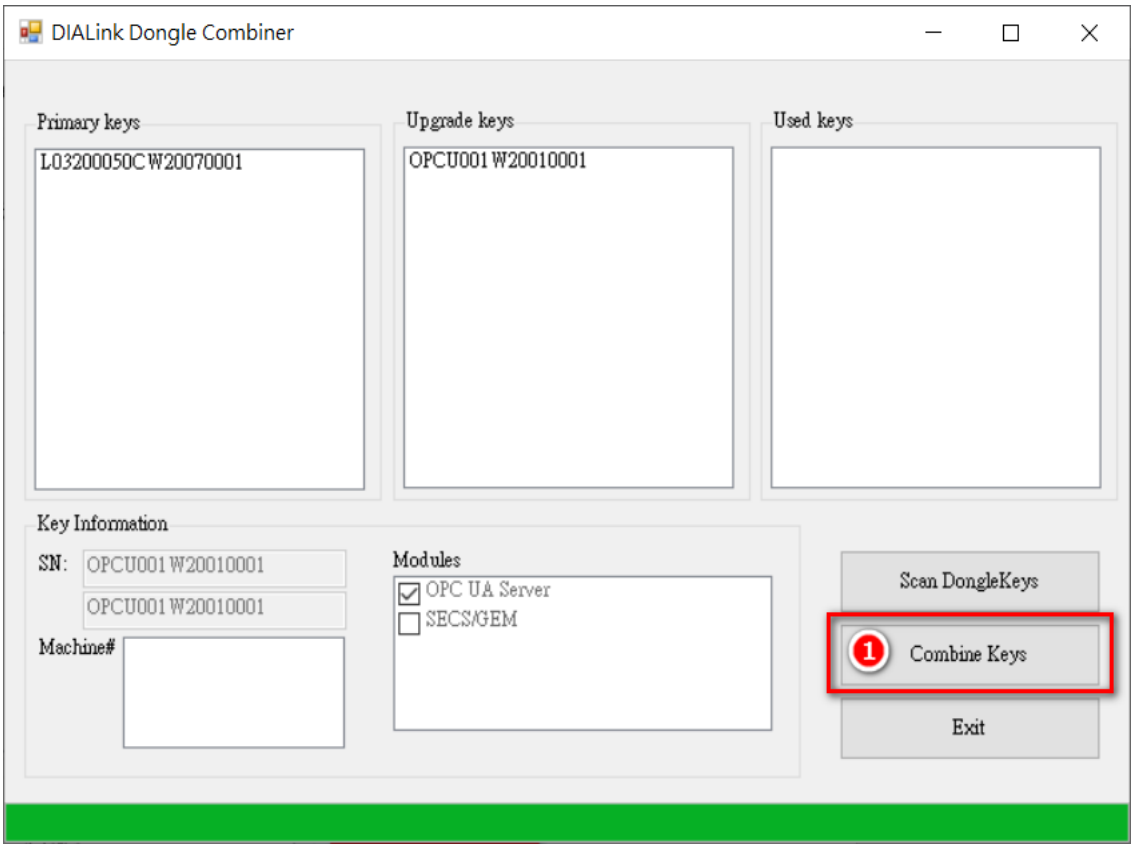
Step 3：插入升级 Key

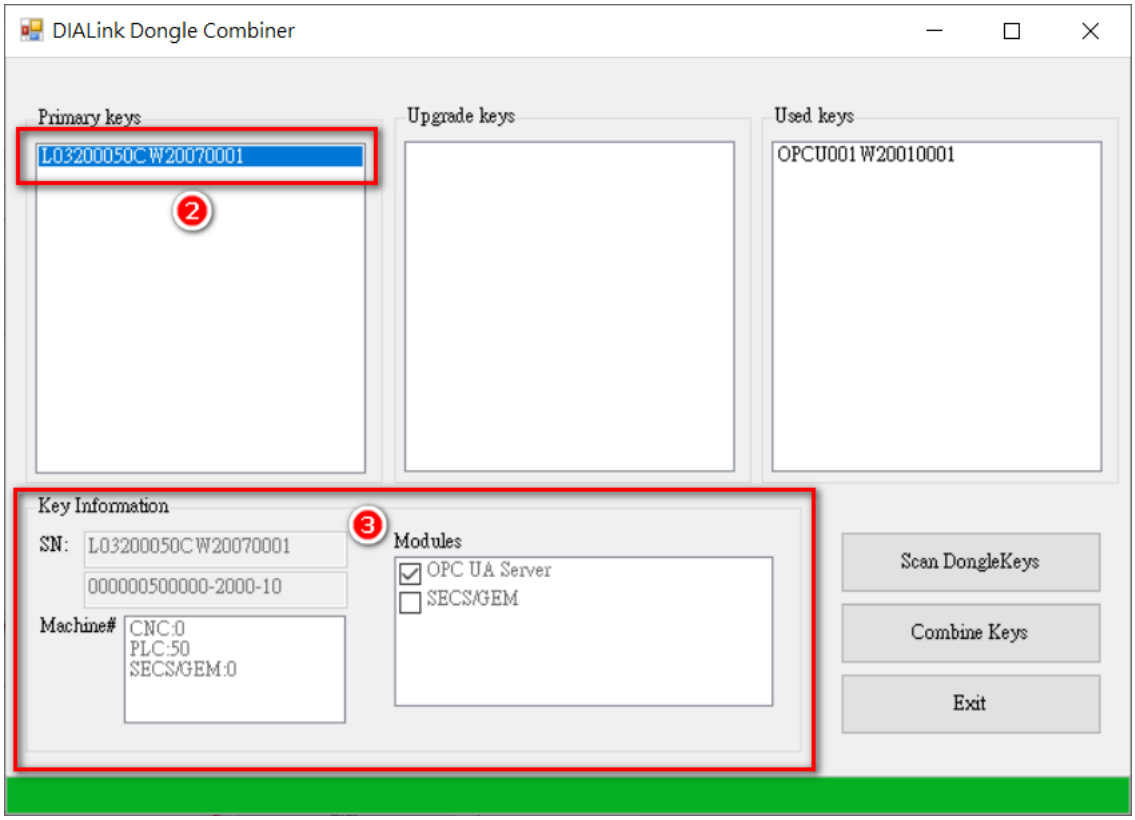
插入升级 Key，❶按下【Scan DongleKeys】显示于❷窗口【Upgrade Keys】内，单击后于❸【Key Information】区域内显示相对应升级设备数及功能模块内容。

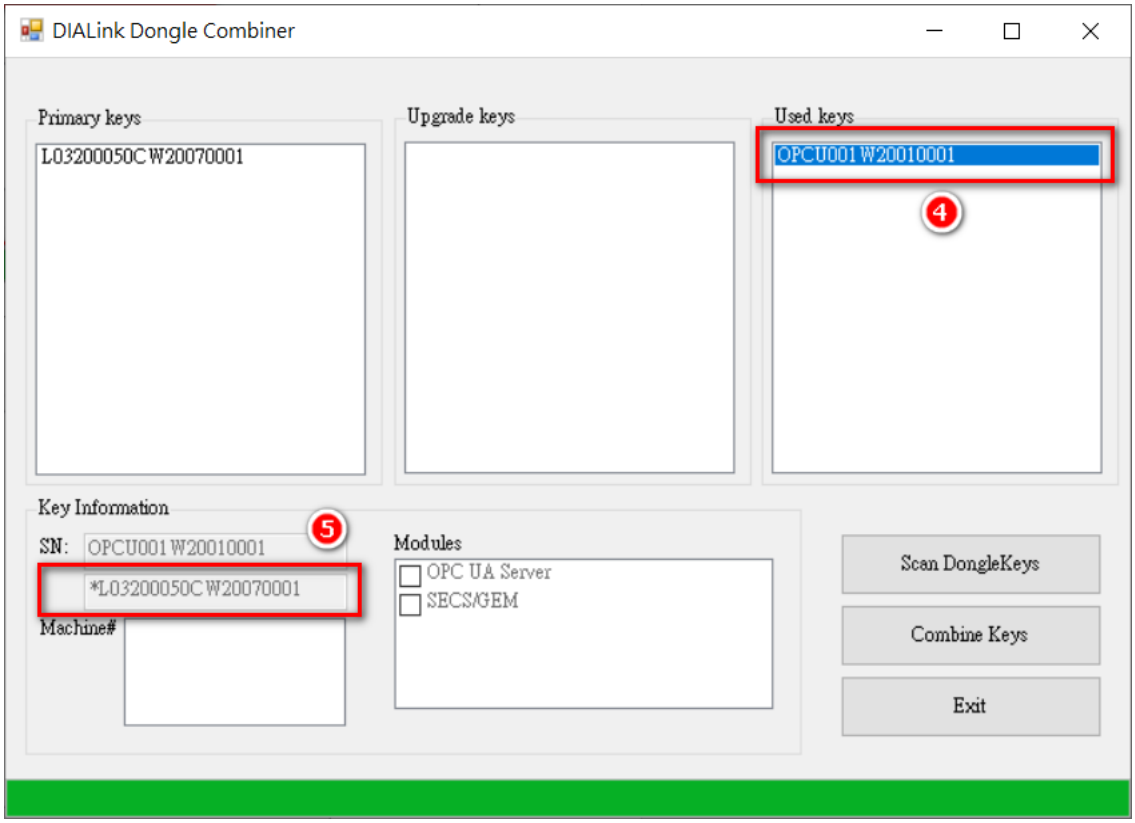


F.2.2 合并 Dongle Key

- 执行工具程序并扫描目前主 Key 和升级 Key 后，确认信息无误后，按下❶【Combine Keys】确认进行合并动作，完成后于【Primary Keys】选择❷并显示合并后的信息❸于【Key Information】。
- 升 Key 执行合并后，会显示于❹，接着单击该项目后，系统将其显示为 Used Key，并记录于❺表示不再使用。
- 设备数升级和模块功能合并 key 后，不需重启 DIALink 服务就可使用。


DIALink Dongle Combiner
Primary keys
L03200050C W20070001
Upgrade keys
OPCU001 W20010001
Used keys
Key Information
SN:
OPCU001 W20010001
OPCU001 W20010001
Machine#
Modules
OPC UA Server
SECS/GEM
Scan DongleKeys
1
Combine Keys
Exit


DIALink Dongle Combiner
Primary keys
L03200050C W20070001
2
Upgrade keys
Used keys
OPCU001 W20010001
Key Information
3
SN:
L03200050C W20070001
000000500000-2000-10
Machine#
CNC:0
PLC:50
SECS/GEM:0
Modules
OPC UA Server
SECS/GEM
Scan DongleKeys
Combine Keys
Exit


DIALink Dongle Combiner
Primary keys
L03200050C W20070001
Upgrade keys
Used keys
OPCU001 W20010001
4
Key Information
SN:
OPCU001 W20010001
5
*L03200050C W20070001
Machine#
Modules
OPC UA Server
SECS/GEM
Scan DongleKeys
Combine Keys
Exit

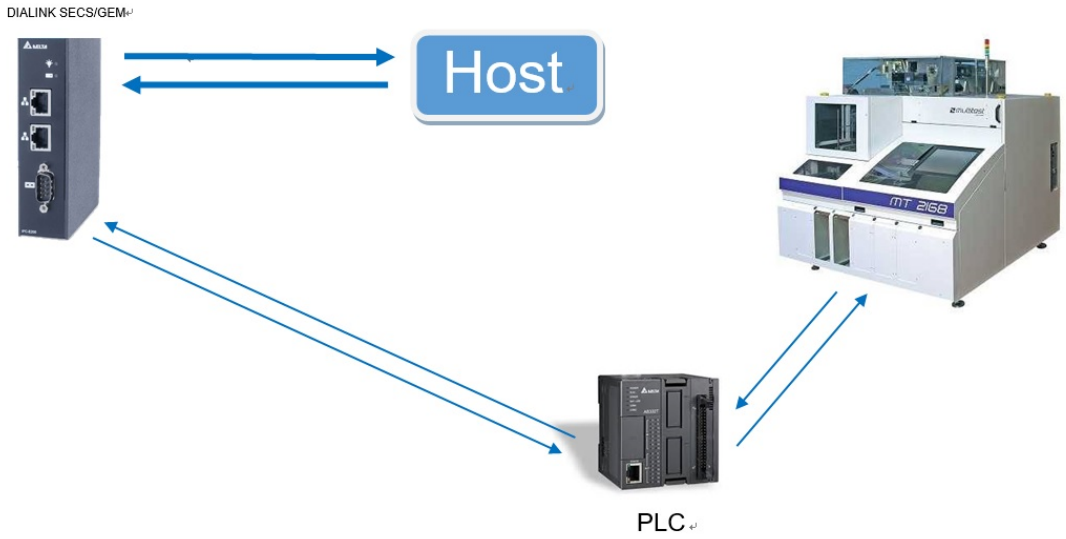
附录G SECS/GEM 用户指南

目录

G.1 架构说明 G-2
G.2 安装及操作流程表 G-2
G.3 SECS/GEM 设定说明 G-4
G.3.1 新增 SECS/GEM 设备 G-4
G.3.2 SECS/GEM 设定 G-5
G.3.2.1 SECS/GEM Application G-5
G.3.2.2 PLC 连线参数设定 G-6
G.3.2.3 I/O Mapping Function Block G-7
G.3.2.4 Alarm Manager G-8
G.3.2.5 Event Manager G-9
G.3.2.6 SV DATA G-10
G.3.2.7 DV DATA G-11
G.3.2.8 EC DATA G-12
G.3.2.9 Link V Setting G-13
G.3.2.10 Command Parameter G-14
G.3.2.11 Remote Command G-15
G.3.2.12 Spool Allow G-17
G.3.2.13 Parameter Manager G-18
G.3.2.14 PPID Mapping Manager G-19
G.3.2.15 Formatted Process Parameter G-20
G.3.2.16 Formatted Command Code G-21
G.3.2.17 Unformatted Process Parameter G-22
G.3.3 联机 G-23
G.3.4 事件 G-25
G.3.5 资讯 G-26
G.3.6 Recipe G-26
G.3.7 Operation G-27
G.3.8 修改 LOG 的进阶设定 G-27
G.4 错误排除 G-28

G.1 架构说明

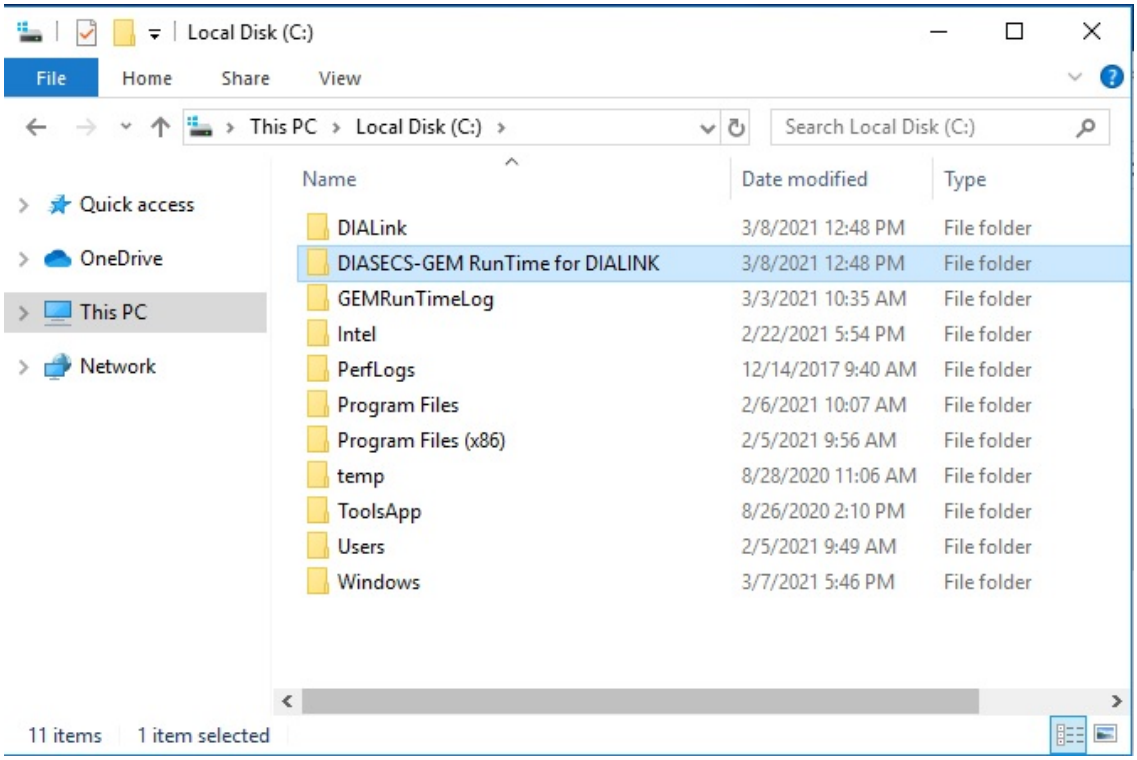
DIALINK SECS/GEM 模块的主要功能为：将不具备 SECS/GEM 的老旧设备，通过 PLC 将数据以 SECS/GEM 的通讯流程上报给上位系统，让旧设备也能提升到得以让上位系统监控与控制。DIALINK SECS/GEM 模块通讯架构如下图，可以跟具有 SECS/GEM 通讯能力的 Host (上位系统，此 Host 可能是 MES) 通讯，DIALINK SECS/GEM 模块通过 PLC 传递 Non-SECS 设备的各种数据，可以把指令、Recipe 等信息传送给设备，也可以将设备的各种数据上报给上位系统



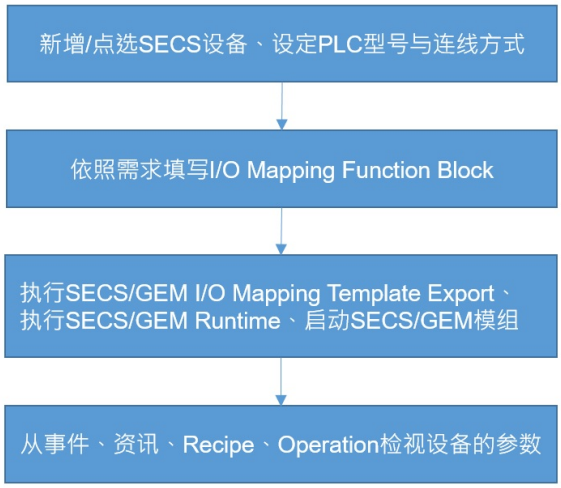
G.2 安装及操作流程表

只要安装了 DIALink1.4 以上的版本，不论使用【完全安装】，还是选择【DIALink only】都会将 SECS/GEM 模块安装在系统中，但是只有购买了模块权限的用户才能建立 SECS/GEM 类型的设备。

安装过程中会新增 SECS/GEM 模块，SECS 的配置文件与执行程序在 C:\\DIASECS-GEM RunTime for DIALink 文件夹中。



- 使用 SECS 功能的流程顺序，如下：



Step 1: 【设备总览】:新增 SECS 设备 >【PLC 连线参数设定】:选择 PLC 型号并设定 IP 地址及 Port

Step 2: 【I/O Mapping Function Block】: 依序设定各种 Function 所要配置的寄存器种类与地址，此设定需与实际设备如何将信息写入 PLC，或从 PLC 取得 Recipe 或命令的设定一致，否则通讯信息会错乱。

Step 3: 【SECS/GEM Application】: 执行 SECS/GEM I/O Mapping Template Export 及 SECS/GEM Runtime。(模组主程序)。

Step 4: 联机后，可从事件、信息、Recipe 及 Operation 检视设备的参数。

G.3 SECS/GEM 设定说明

G.3.1 新增 SECS/GEM 设备

- 新增设备

登入 DIALink 后，依次单击❶左方【设备总览】 > ❷【新增一台设备】，或者单击右上方➕符号。





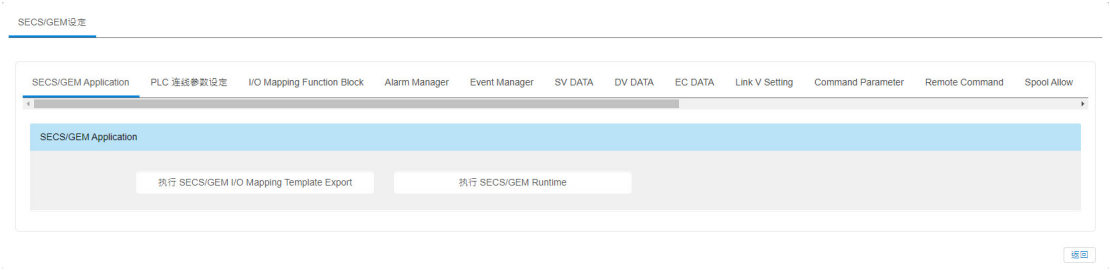
【注】新增 SECS 设备需要有 SECS 授权才能新增。若新增 SECS 设备的数量超过授权数量，则无法新增。

【注】SECS 设备与其它设备不同。其它的设备，一般都是用 DIALink 的核心直接开启与设备通讯的管道，如：直接与 CNC 或 PLC 设备沟通。而 SECS 模块则是需要一个中间媒介 SECS/GEM Runtime，它是用 SECS/GEM 的通讯格式来跟第三方的 HOST 沟通，以及通过 PLC 跟机台设

备沟通，因此，当新增 SECS 设备后，即便连线参数都设定正确，网页上的 Device 也不会自动联机及自动取得点位，反而是需要通过设定后，并手动将 SECS/GEM Runtime 启动，让它与上位 HOST 及下位设备沟通。

G.3.2　SECS/GEM 设定

G.3.2.1 SECS/GEM Application



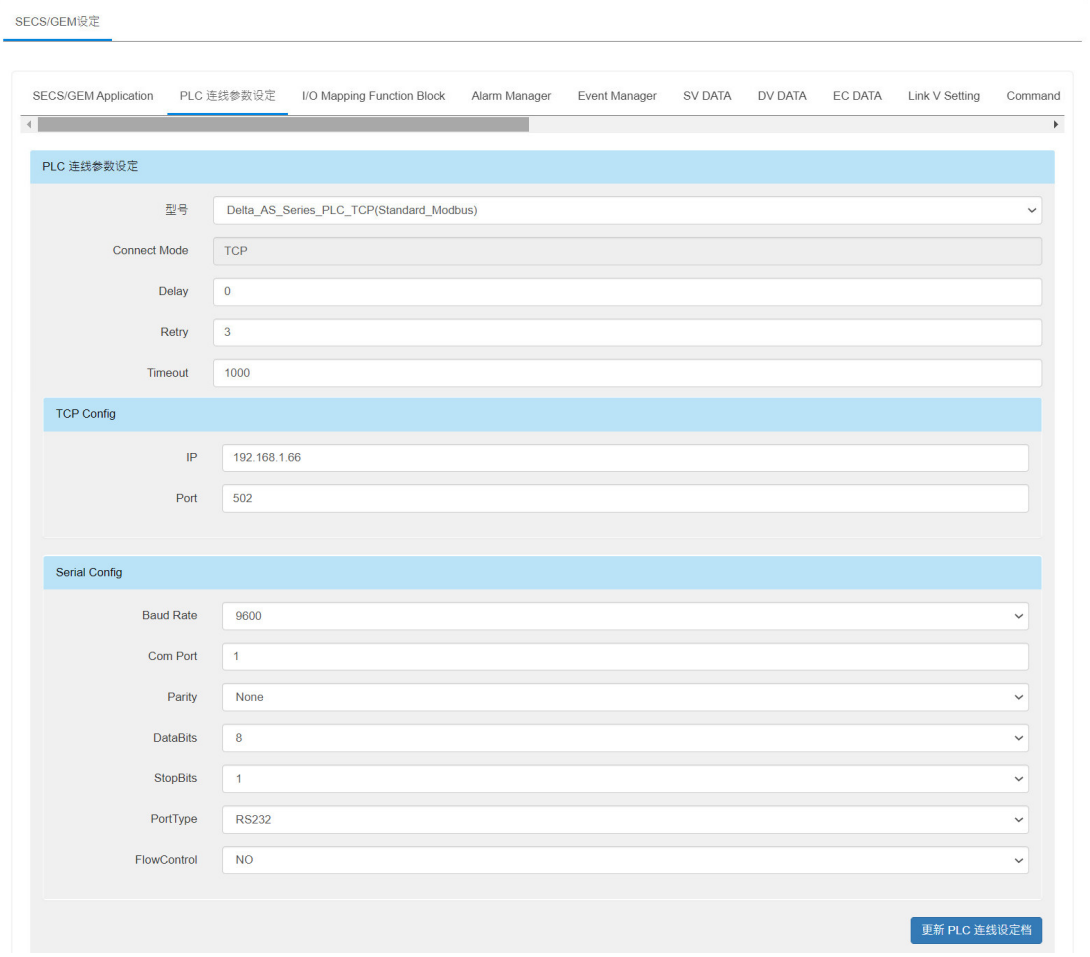
- 执行 SECS/GEM I/O Mapping Template Export：当用户对 I/O Mapping Function Block、Alarm Manager、Event Manager、SV DATA、DV DATA、EC DATA、Link V Setting、Command Parameter、Remote Command、Spool Allow、Parameter Manager、PPID Mapping Manager、Formatted Process Parameter、Formatted Command Code、Unformatted Process Parameter 等页面进行新增、删除、修改后，需执行 “执行 SECS/GEM I/O Mapping Template Export” 钮，以确保设定有写入 Config 文件。

- 执行 SECS/GEM Runtime：执行 Runtime 后，即可进入联机状态。详细内容于 G.3.3 联机章节，进行介绍。

【注】执行 Export 或 Runtime 时，由于后端执行时设定不同，以及系统资源状况不同，需花费大约 10~50 秒不等的时间，故点选执行按钮后请等待。若执行成功，右上角有绿色框提示。反之，若执行失败则有红色框提示。当执行超过 50 秒为非预期情况时，网页将回到原页面并显示 TimeOut，请重新执行一次确认是否成功。

【注】在执行 Runtime 的过程中，若于等待页面执行 F5（Refresh），接着又重新执行 Runtime，此步骤会造成系统错误，系统会跳出错误信息“执行 SECS/GEM Runtime 失败”。回复动作：回到【设备总览】，此时 SECS 设备可能已联机成功，若无联机成功，则再执行一次 Runtime 即可再次联机。

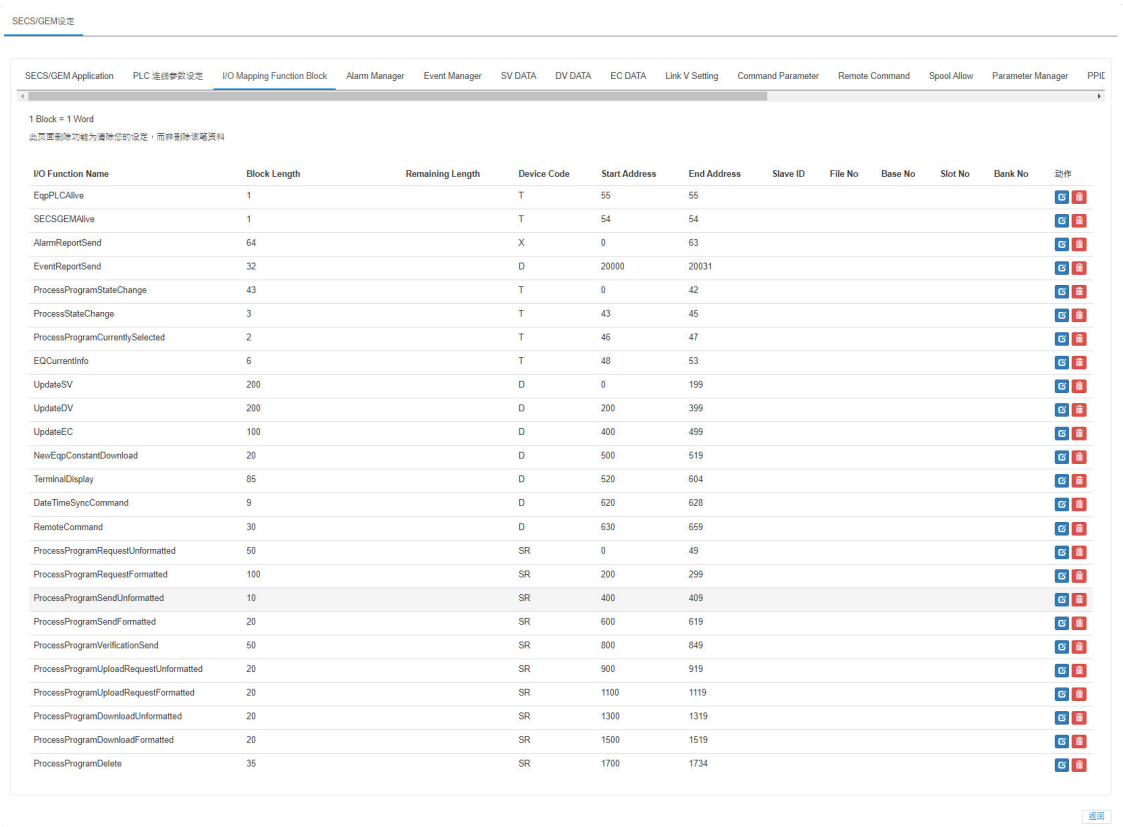
G.3.2.2 PLC 连线参数设定



- PLC 连线参数设定：选择所要设定的型号（必选），其它字段按照需求填入。
- TCP Config：设定 IP 及 Port（必填），其它字段按照需求填入。
- Serial Config：相关字段按照需求填入。

【注】若型号、IP 及 Port 未设定的情况下，则无法顺利启动 Runtime，系统将显示“PLC driver initial failure”等错误信息。

G.3.2.3 I/O Mapping Function Block



I/O Mapping Function Block 寄存器只能使用 WORD 寄存器，其它 Bit、Double Word 寄存器都不能被 SECS 使用。不同的 PLC 会有不同的寄存器范围与特性，需参考各家 PLC 的文件，针对各机种设定可用的寄存器种类与范围。如：使用 Delta DVP 系列的 PLC，只能使用 T、C、D 这三种寄存器。而 Delta AS 系列的 PLC，则是使用 X、Y、D、SR、E、T、C 的寄存器。在设定 Address 时则不能超过各寄存器的范围。

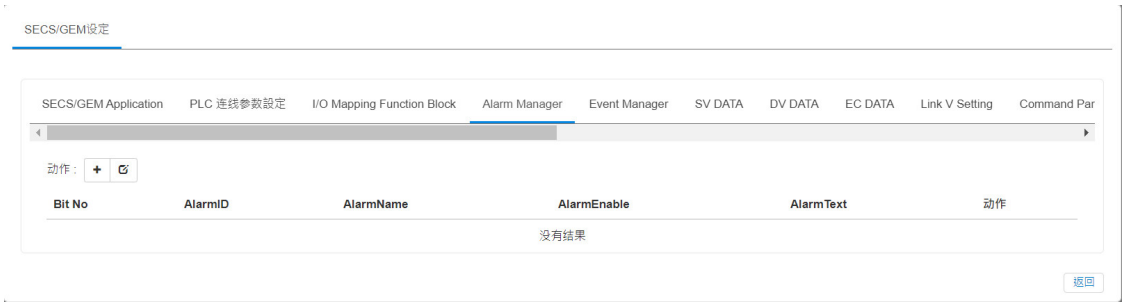
另设定 Device Code 时，系统会列出目前所使用 PLC 的可用寄存器。Start Address 则是设定欲使用的区段起始点，并针对该区段默认的 Block（Word）设定 End Address。如: Start Address 设定 0，该区段默认为 200 个 Block(Word)，系统则显示使用 D0～D199 的位置。需特别注意，Device Code（Range）会取决于所选定的 PLC 设备，若当 Address 超出范围，用户可重新定义 Block Length，再重新设定 Start Address。

Function 若以【能否修改 Block Length】来区分则可分为两类，Function 若不允许用户修改 Block Length，则其 Remaining Length 一律呈现为 0，而 Function 允许用户修改 Block Length 的，其 Remaining Length 则会显示可用的、剩余的 Block 有多少。第二类 Function 当设备使用的信息量超过默认的 Length 时，可进行变更。虽然 Remaining Length 是该 Function 在经过配置之后剩余的空间，但因为有些 Function 有保留部分 Block 给系统使用，让某些 Remaining Length 总是比计算后的还少几个 Block，此为正常现象。

【注】若要对 Alarm Manager、Event Manager、SV DATA、DV DATA、EC DATA、Remote Command、

PPID Mapping Manager、Formatted Command Code 及 Unformatted Process Parameter 进行新增、删除及修改等动作，则需先在 I/O Mapping Function Block 页面进行相关设定，各功能页面也会提示需要设定的项目。如：单击 Alarm Manager，则跳出“I/O Mapping Function Block 尚未设定，无法在本页面进行设定。请先完成 I/O Mapping Function Block：AlarmReportSend”。

G.3.2.4 Alarm Manager

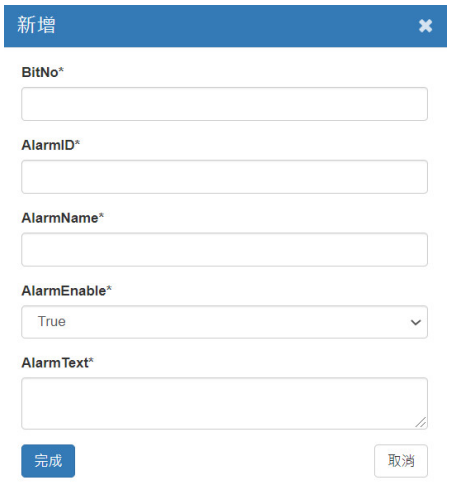


网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。

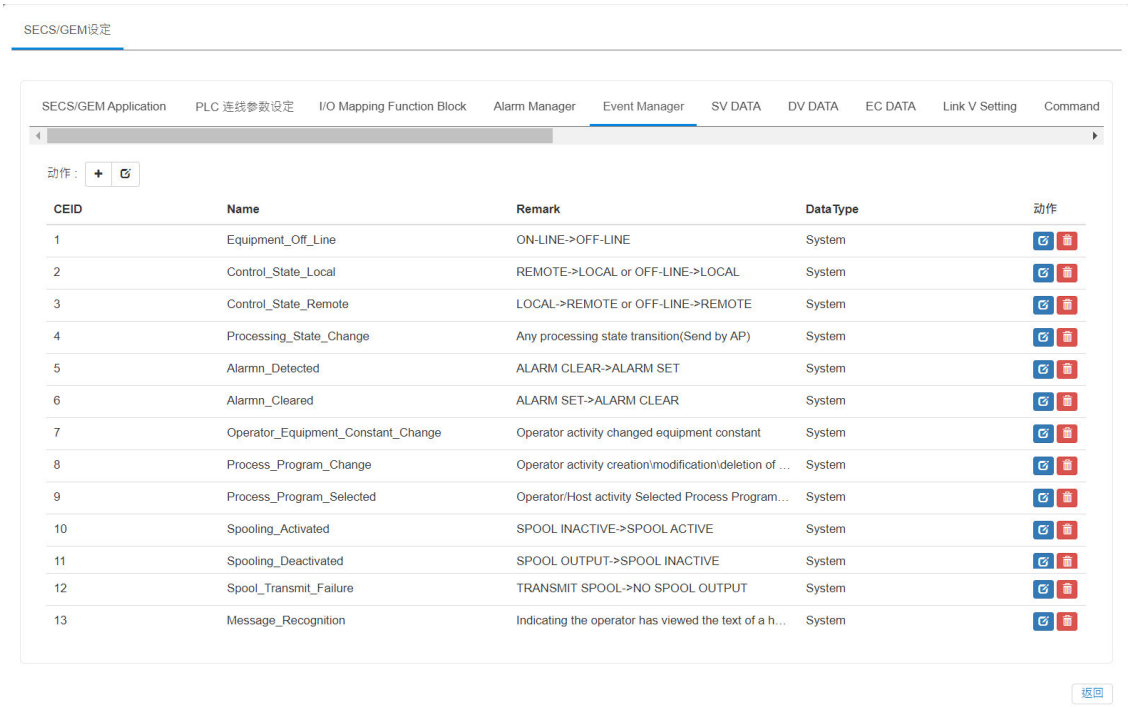
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 Alarm Manager

单击动作 ✚，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.5 Event Manager



Event Manager 分成 System 与 Eqp 两种，System 是从文件读取出来，仅可以修改 CEID 字段数据，无法删除。而 Eqp 则可让用户进行新增、删除及修改。

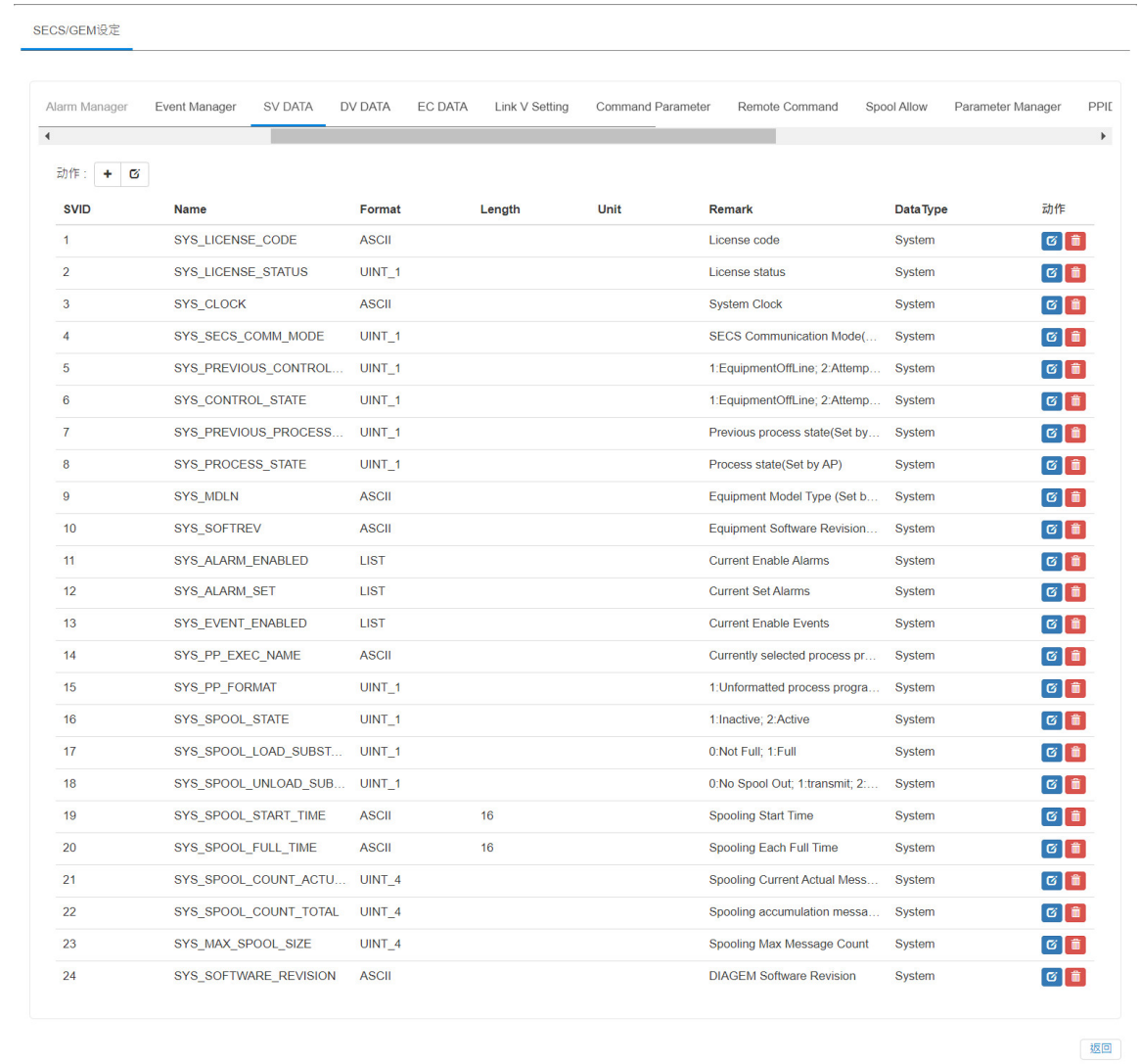
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 Event Manager

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.6 SV DATA



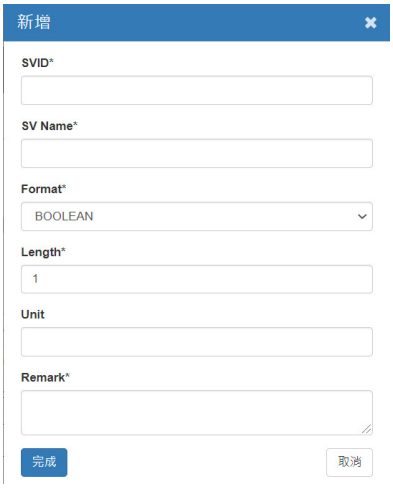
SVID	Name	Format	Length	Unit	Remark	Data Type	动作
1	SYS_LICENSE_CODE	ASCII			License code	System	
2	SYS_LICENSE_STATUS	UINT_1			License status	System	
3	SYS_CLOCK	ASCII			System Clock	System	
4	SYS_SECS_COMM_MODE	UINT_1			SECS Communication Mode(...	System	
5	SYS_PREVIOUS_CONTROL...	UINT_1			1:EquipmentOffLine; 2:Attemp...	System	
6	SYS_CONTROL_STATE	UINT_1			1:EquipmentOffLine; 2:Attemp...	System	
7	SYS_PREVIOUS_PROCESS...	UINT_1			Previous process state(Set by...	System	
8	SYS_PROCESS_STATE	UINT_1			Process state(Set by AP)	System	
9	SYS_MDLN	ASCII			Equipment Model Type (Set b...	System	
10	SYS_SOFTREV	ASCII			Equipment Software Revision...	System	
11	SYS_ALARM_ENABLED	LIST			Current Enable Alarms	System	
12	SYS_ALARM_SET	LIST			Current Set Alarms	System	
13	SYS_EVENT_ENABLED	LIST			Current Enable Events	System	
14	SYS_PP_EXEC_NAME	ASCII			Currently selected process pr...	System	
15	SYS_PP_FORMAT	UINT_1			1:Unformatted process progra...	System	
16	SYS_SPOOL_STATE	UINT_1			1:Inactive; 2:Active	System	
17	SYS_SPOOL_LOAD_SUBST...	UINT_1			0:Not Full; 1:Full	System	
18	SYS_SPOOL_UNLOAD_SUB...	UINT_1			0:No Spool Out; 1:transmit; 2:...	System	
19	SYS_SPOOL_START_TIME	ASCII	16		Spooling Start Time	System	
20	SYS_SPOOL_FULL_TIME	ASCII	16		Spooling Each Full Time	System	
21	SYS_SPOOL_COUNT_ACTU...	UINT_4			Spooling Current Actual Mess...	System	
22	SYS_SPOOL_COUNT_TOTAL	UINT_4			Spooling accumulation messa...	System	
23	SYS_MAX_SPOOL_SIZE	UINT_4			Spooling Max Message Count	System	
24	SYS_SOFTWARE_REVISION	ASCII			DIAGEM Software Revision	System	

SV DATA 分成 System 与 Eqp 两种，System 是从文件读取出来，仅可以修改 SVID 字段数据，无法删除。而 Eqp 则可让用户进行新增、删除及修改。其中，SVID 不得与 DVID、ECID 重复，否则将无法新增成功，也无法让 Runtime 执行成功。

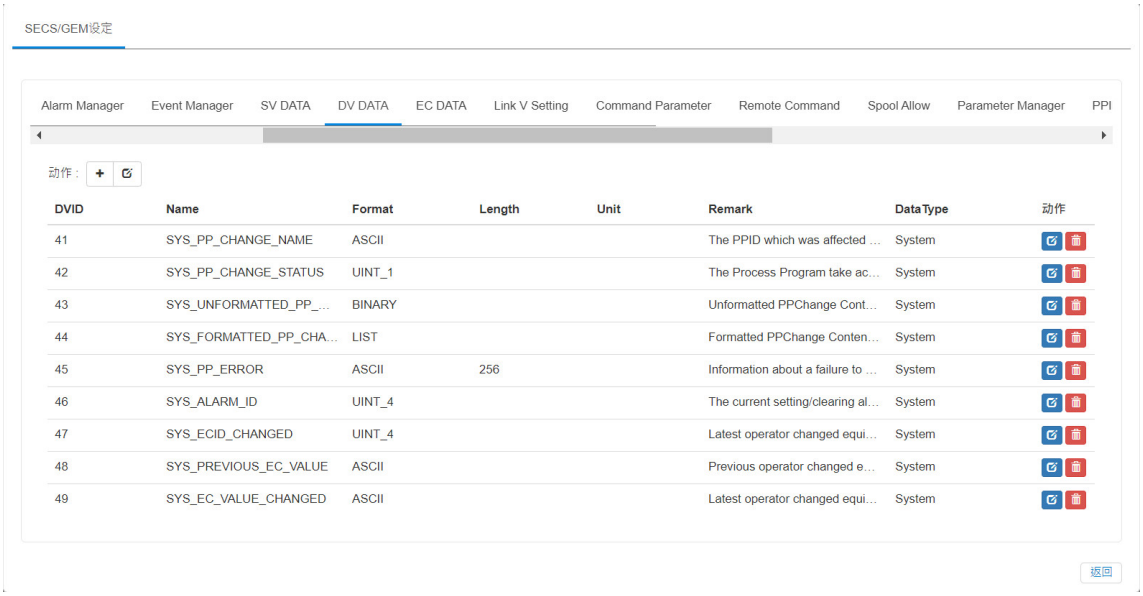
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 SV DATA

单击动作 ✚ ，系统将跳出新增窗口，在相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.7 DV DATA

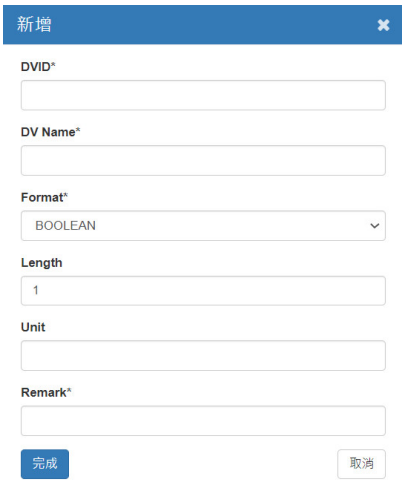


DV DATA 分成 System 与 Eqp 两种，System 是从文件读取出来，仅可以修改 DVID 字段数据，无法删除。而 Eqp 则可让用户进行新增、删除及修改。其中，DVID 不得与 SVID、ECID 重复，否则将无法新增成功，也无法让 Runtime 执行成功。

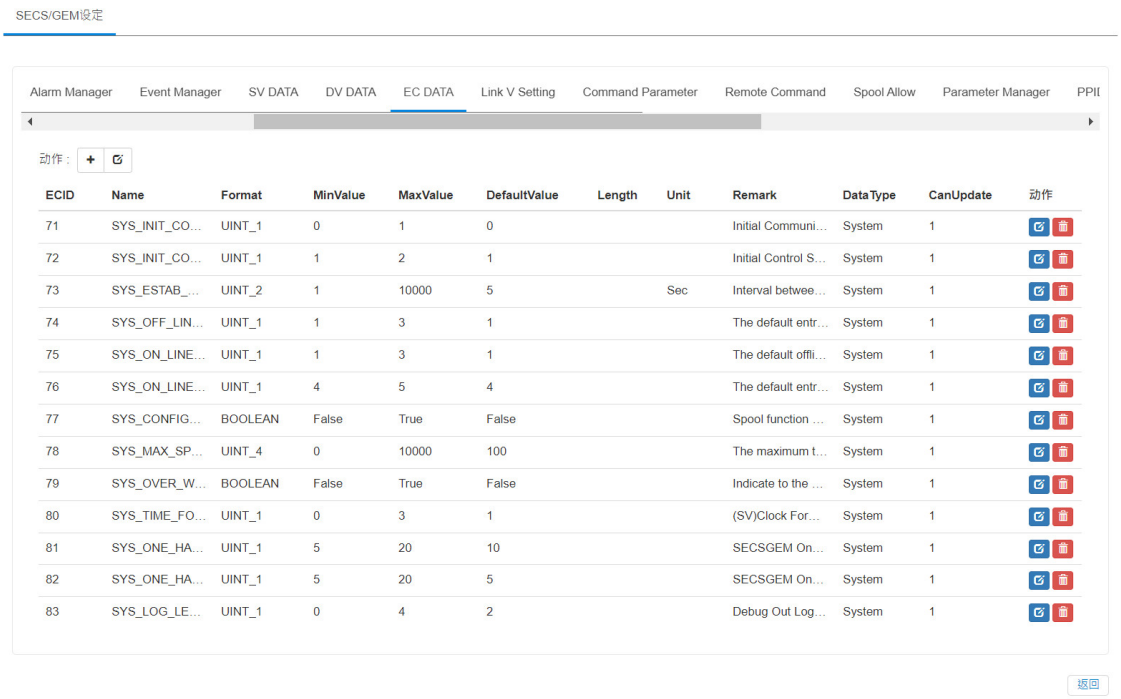
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 DV DATA

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.8 EC DATA

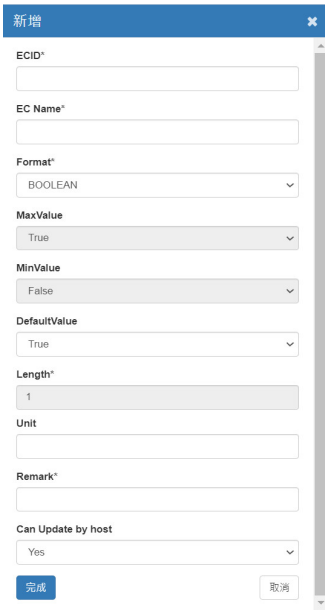


EC DATA 分成 System 与 Eqp 两种，System 是从文件读取出来，仅可以修改 ECID 字段数据，无法删除。而 Eqp 则可让用户进行新增、删除及修改。其中，ECID 不得与 SVID、DVID 重复，否则将无法新增成功，也无法让 Runtime 执行成功。而 EC 数据字段的 DefaulValue 则必须介于 MaxValue 与 MinValue 之间。

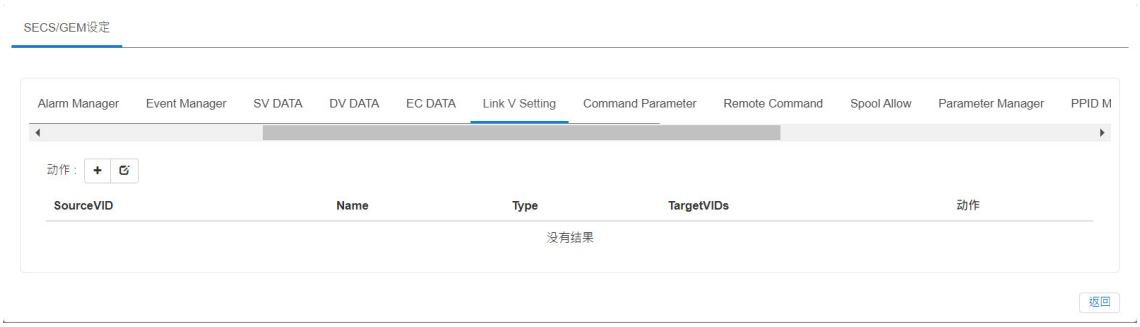
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 EC DATA

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.9 Link V Setting



网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。此页面是用来将 SV、DV、EC 作群组绑定的用途。其中，若将 SV、DV 的 Format 设定为 Link 时，才能当成 Source 来绑定其它数据。若原绑定的 SV、DV、EC 的 VID 有被修改过或数据删除，那么该笔绑定的记录则会自动被删除，用户需重新新增并再次绑定。

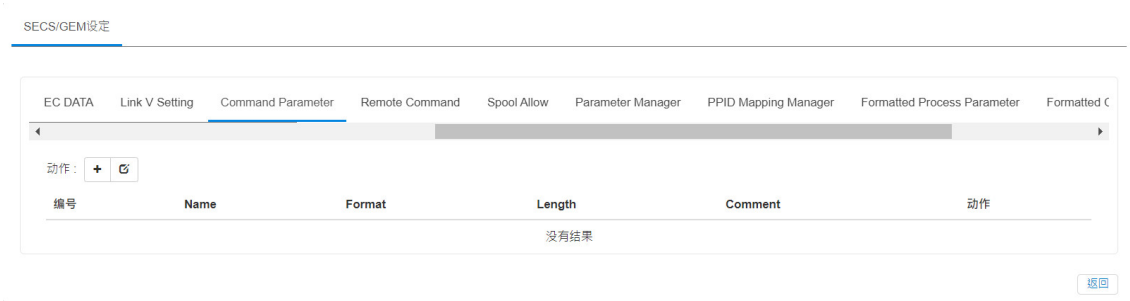
【注】不能互相绑定（如 A 绑 B、B 绑 A），否则执行 Export 及 Runtime 时会出现错误信息。

- 新增 Link V Setting

单击动作 ✚ ，系统将跳出新增窗口，选择所要绑定主 VID，并勾选欲绑定的 VID，即可新增成功。（每一笔主 VID 可绑定多个 VID，但最多只能绑定 200 个 VID）



G.3.2.10 Command Parameter



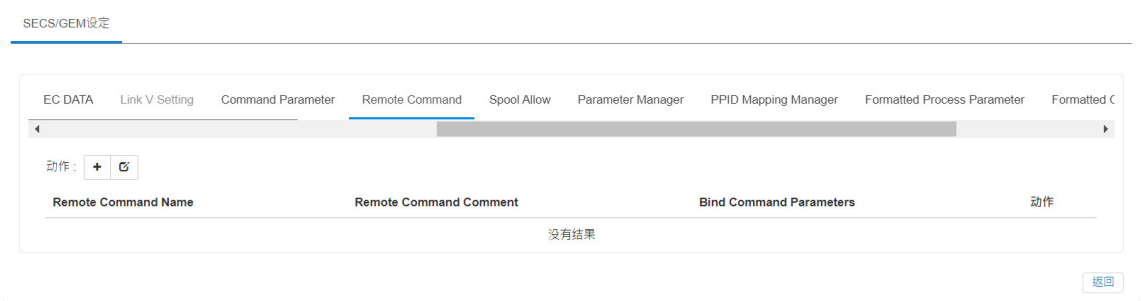
网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。

- 新增 Command Parameter

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.11 Remote Command



网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。若原绑定的 Command Parameter 的 Name 有被修改过，那么此笔记录则会自动被删除，用户需重新新增并再次绑定。

【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

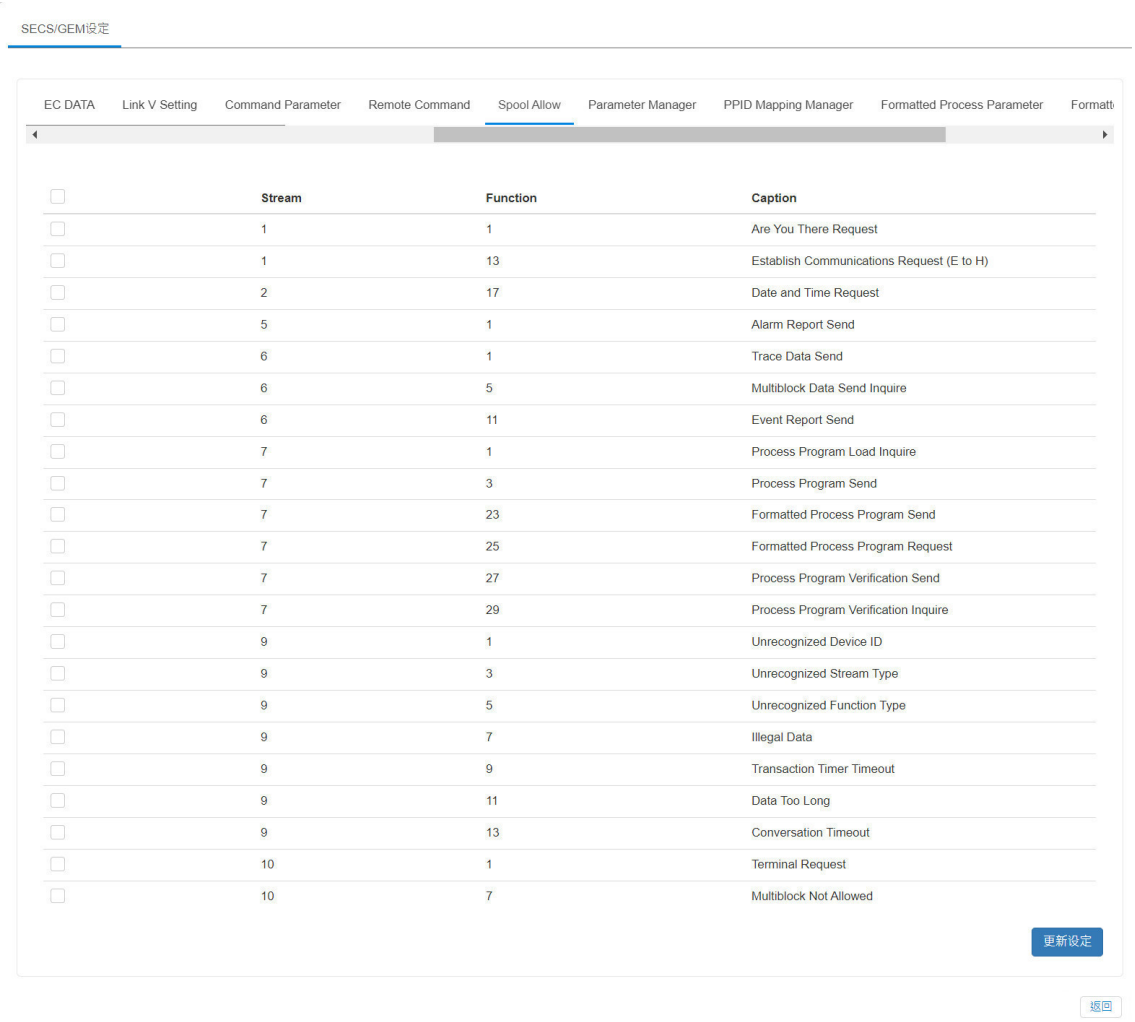
- 新增 Remote Command

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功) (可绑定多个 Command Parameters)

【注】新增 Remote Command 前，需先新增 Command Parameter，才能勾选 Bind Command Parameters。

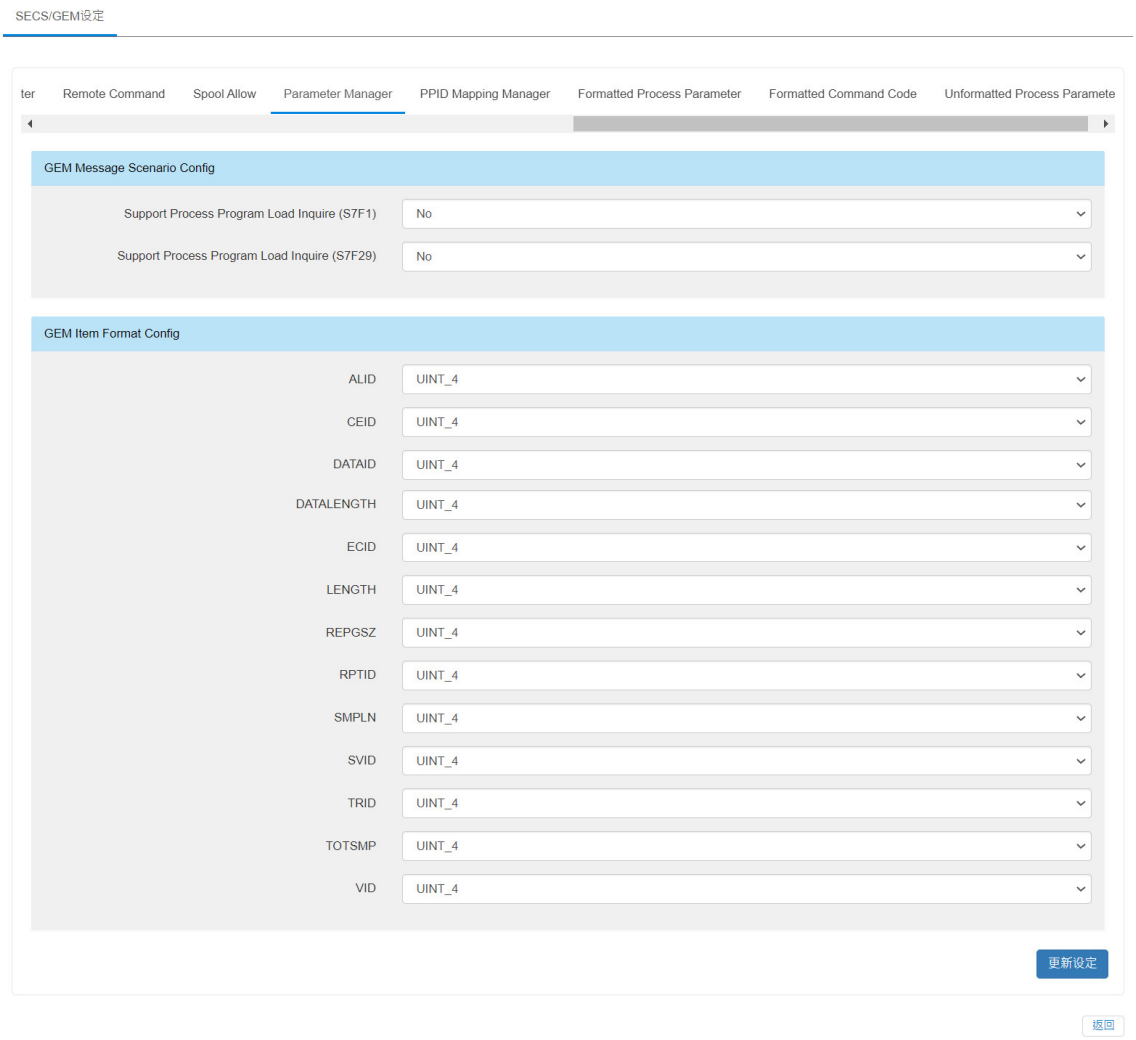

新增
Remote Command Name*
Remote Command Comment*
完成
取消
选择 Bind Command Parameters
Command Name
Format
Length
Comment
没有结果

G.3.2.12 Spool Allow



网页后端程序将文件读出，提供给前端网页作设定，勾选设定后单击【更新设定】，系统会将设定结果储存到文件中。(Spool Allow 为选填)

G.3.2.13 Parameter Manager



网页后端程序将文件读出，提供给前端网页作设定，修改设定后并单击【更新设定】，系统会将设定结果储存到文件中。系统依照 Alarm、Event、SV DATA、DV DATA、EC DATA 所设定 ID 的最大值，提供不同的 GEM Item Format Config，而此页面的 SVID 是指 SVID 及 DVID 的设定，卡控规则如下：

- UINT_1：ID 卡控范围为 1~255
- UINT_2：ID 卡控范围为 1~65535
- UINT_4：ID 卡控范围为 1~4294967295
- UINT_8：ID 卡控范围为 1~18446744073709551615

举例来说：当 ALID 设定为 UINT_2，则到 Alarm Manager 页面新增或编辑数据时，Alarm ID 只能设定在 1~65535 之间，当 ALID 设定为 UINT_1，则 Alarm ID 只能设定在 1~255 之间。若 SVID 设定为 UINT_2，则在 SV DATA 与 DV DATA 两个页面新增或编辑的数据，其 ID 只能在 1~65535 之间。

另外，系统针对 VID 的设定卡控规则，如下：

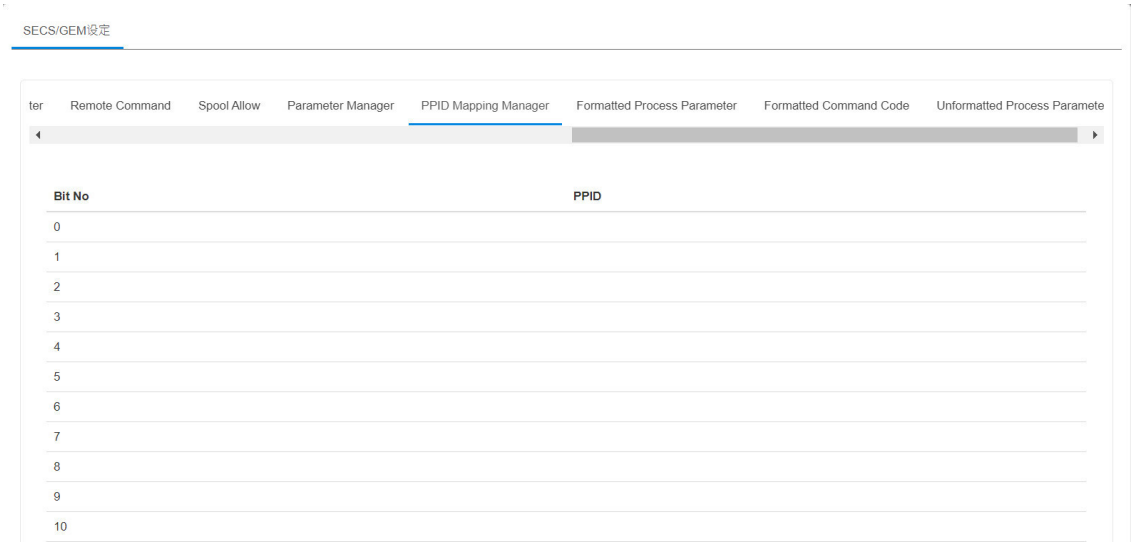
VID 最小格式必须是 SVID 和 ECID 最大的设定格式。

举例 1：SVID（UINT_1）、ECID（UINT_2），故 VID 仅能设定 UINT_2、UINT_4、UINT_8。

举例 2：SVID（UINT_1）、ECID（UINT_4），故 VID 仅能设定 UINT_4、UINT_8。

举例 3：SVID（UINT_8）、ECID（UINT_4），故 VID 仅能设定 UINT_8。

G.3.2.14 PPID Mapping Manager

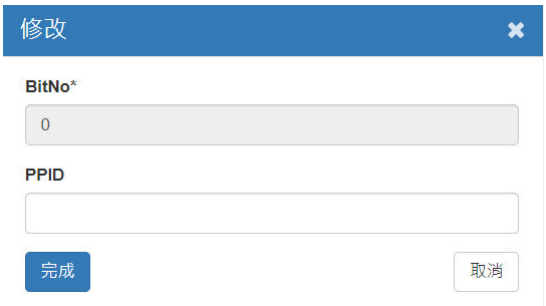


PPID 有特殊规则，为了让 Runtime 读取，故文件内必须有 512 笔数据，其中未使用的内容可留空白。网页后端程序将文件读出，用户可通过网页进行修改，修改设定后会将结果储存到文件中。

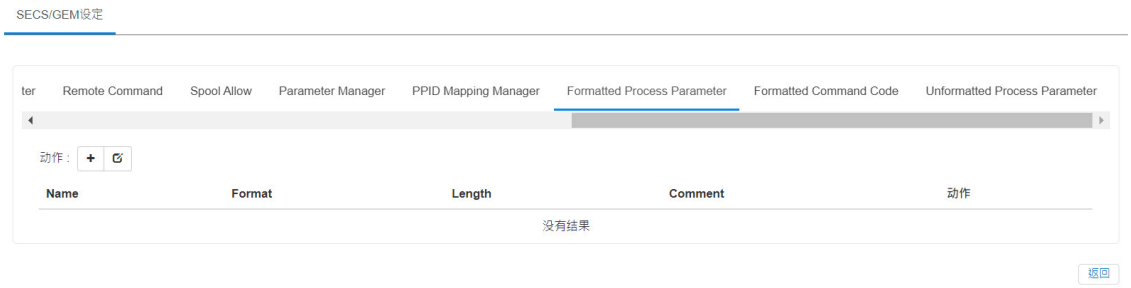
【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 修改 PPID Mapping Manager

  直接单击欲改选的那笔，系统将跳出该笔的修改窗口，于 PPID 字段输入数值或文字。



G.3.2.15 Formatted Process Parameter



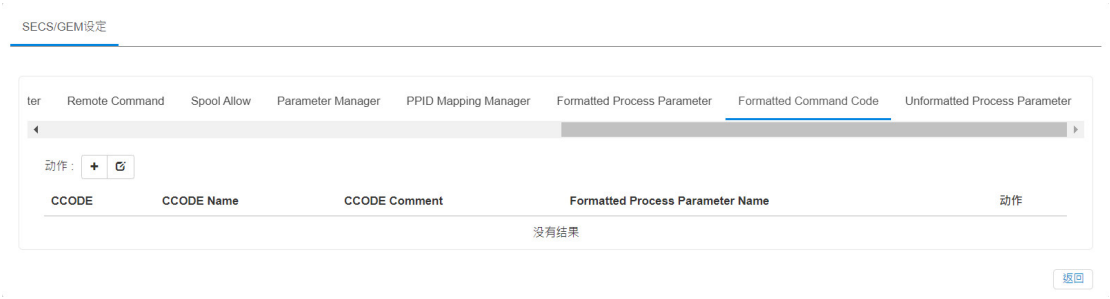
网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。

- 新增 Formatted Process Parameter

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.2.16 Formatted Command Code



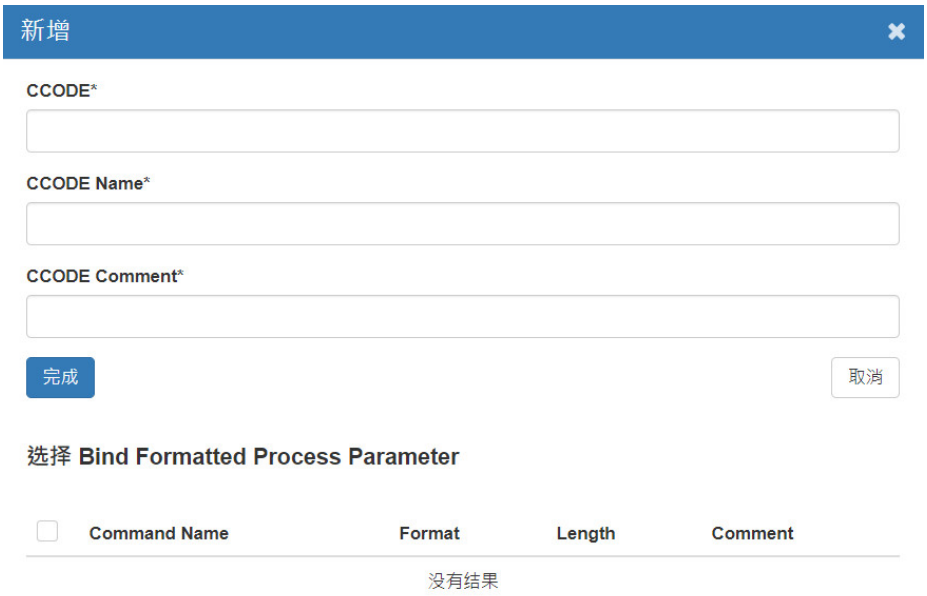
网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。若原绑定的 Formatted Process Parameter 的 Name 有被修改过，那么此笔记录则会自动被删除，用户需重新新增并再次绑定。

【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

- 新增 Formatted Command Code

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)

【注】新增 Formatted Command Code 前，需先新增 Formatted Process Parameter，才能勾选 Bind Formatted Process Parameter。



G.3.2.17 Unformatted Process Parameter



网页后端程序将文件读出，提供给前端网页作设定，设定后将设定结果储存到文件中。用户可进行新增、删除、修改等动作。可设定 Unformatted PPBody Format，此设定为规范 SECS-II Item（PPBODY）之传输格式（用以检查此 SECS-II Item：1. HOST Download 是否与设定一致、2. 将 Eqp 上报 PLC 数据转换成设定格式）

【注】请于 I/O Mapping Function Block 页面设定 Device Code 及 Start Address，设定后才能开启此页面功能。

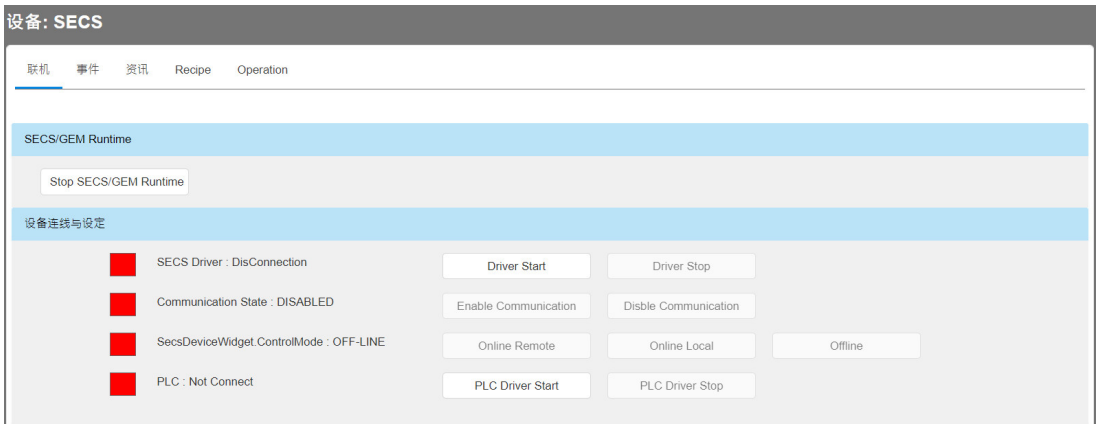
- 新增 Unformatted Process Parameter

单击动作 ✚ ，系统将跳出新增窗口，于相关字段输入数值或文字。(所有必填字段均不可空白，否则无法新增成功)



G.3.3　联机

于【SECS/GEM Application】启动【执行 SECS/GEM Runtime】即可进入如下页面。

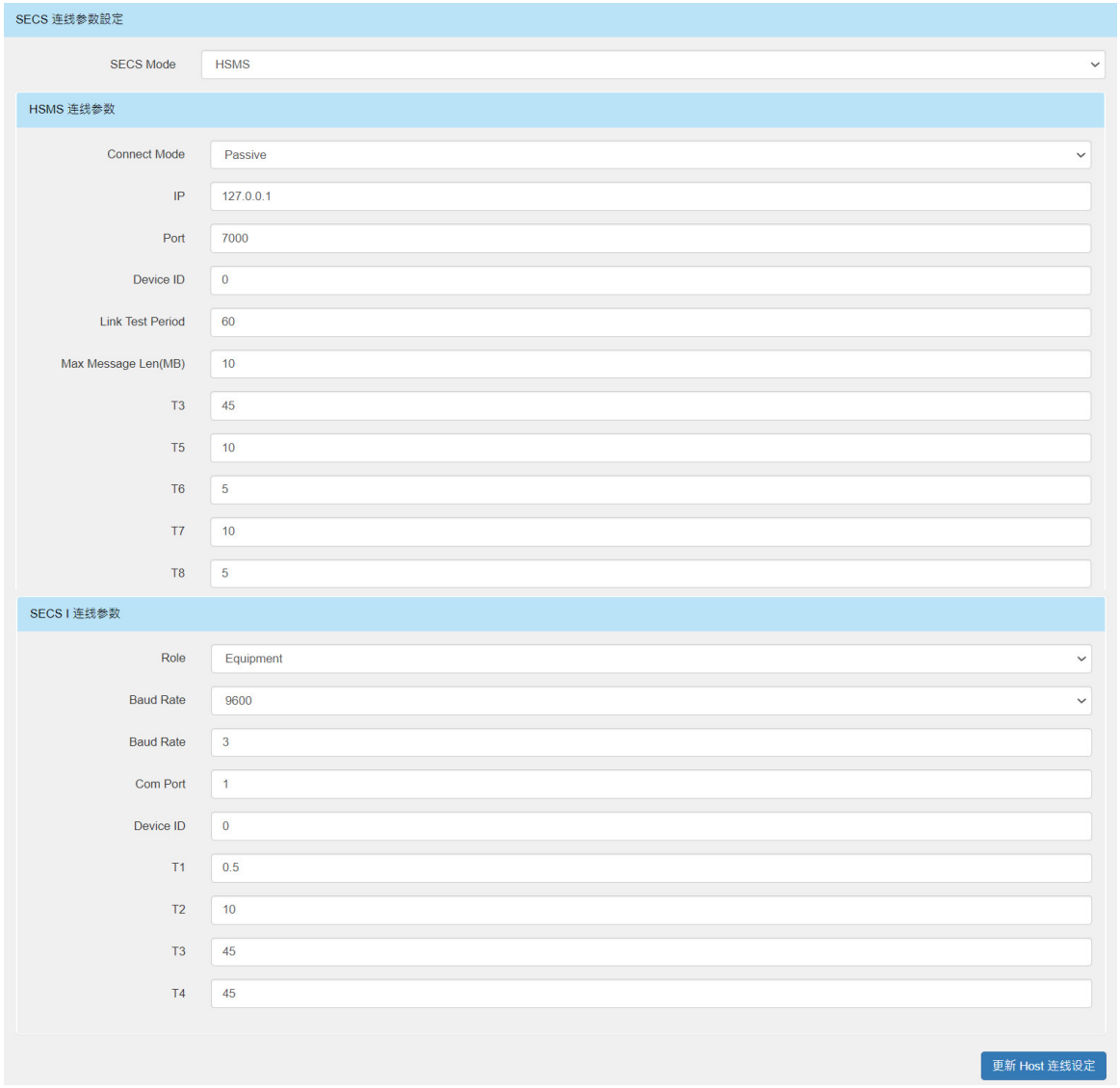


- SECS/GEM Runtime：若要停止联机，可单击“Stop SECS/GEM Runtime”按钮。

- 设备联机控制：先将 SECS driver 启动，当建立联机后才能启动 Communication，接着才能设定 Control mode。而 PLC 则可独立启动。

【注】只有在 SECS Driver 建立完成（绿灯）后，Communication 才能 Enable，若 Driver Stop 则 Communication 也会断线。而 Communication 在 Enable 之后 Control Mode 才能自由切换。另，假使 Control Mode 已设定为 Remote 或 Local，此时将 Communication Disable 造成断线，Control Mode 并不会变成 Offline，且能在 Remote 跟 Local 之间切换，只有手动切换到 Offline（红灯）时，才会无法再切回 Remote 跟 Local。此为 SECS 的正常流程。

【注】PLC Driver Stop 后需等待提示页面消失后再进行 PLC Driver Start，避免当设备尚未中断时又被启动，而跳出错误信息。

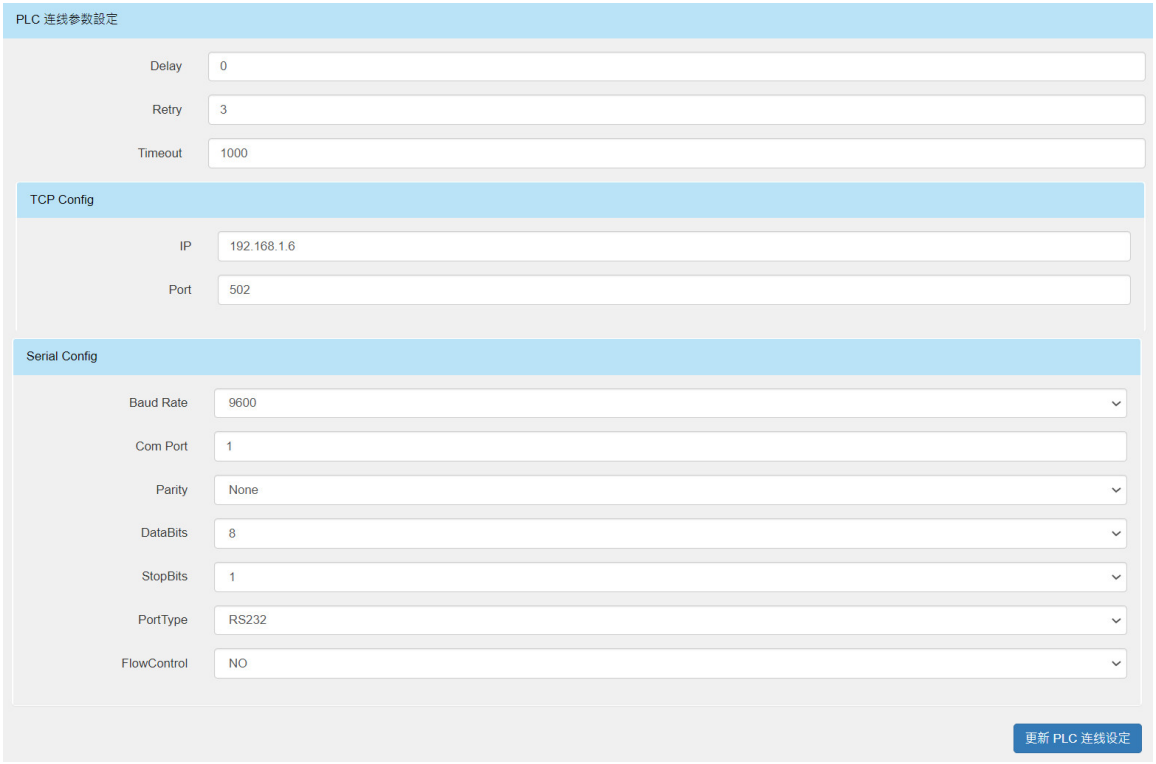


- SECS 连线参数设定：用户可依照需求进行设定。若同时有 2 个以上的 SECS 设备，需特别注意 Port 不能重复，否则会造成 Runtime 无法启动。

  SECS mode: 设定需与 HOST 相同。

  Connect Mode: 则与 HOST 相反(如 :HOST 设定为 Passive ;Connect Mode 则设定为 Active)

【注】当 Driver 联机时，则无法对 SECS 连线参数做设定。若要设定则需将 Driver 停止，方可进行设定。

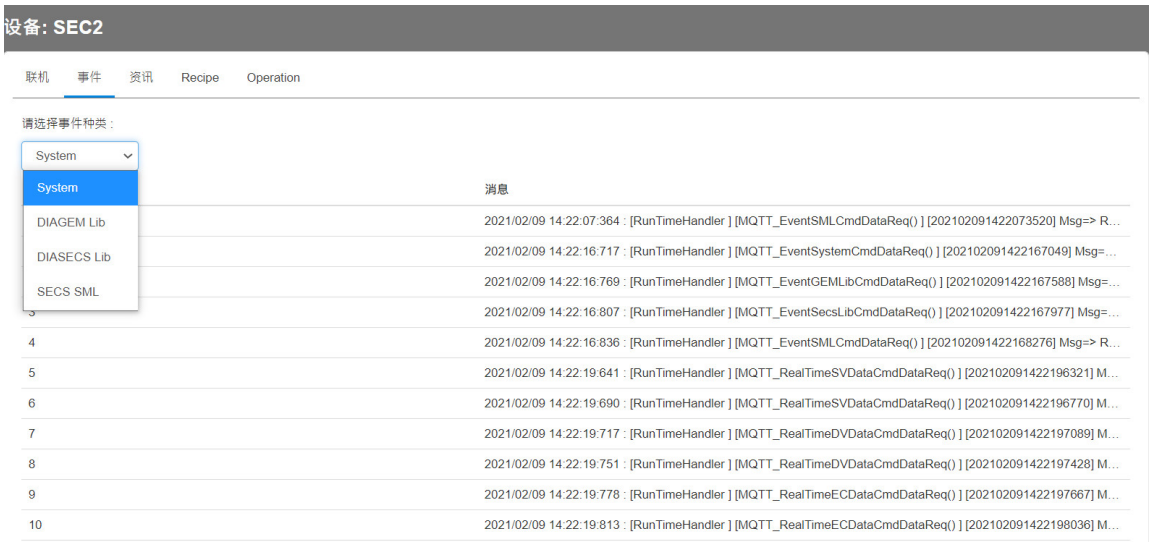


- PLC 连线参数设定：用户可依照需求进行设定。

【注】当 PLC 联机时，则无法对 PLC 连线参数做设定。若要设定则需将 PLC 断线，方可进行设定。

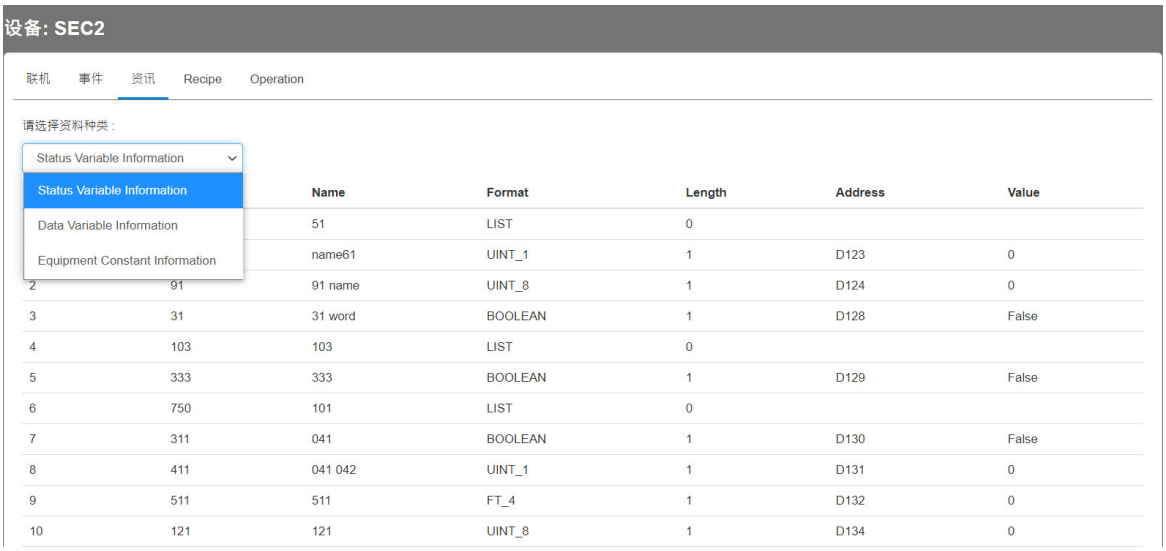
G.3.4　事件

「事件」种类分为: System、DIAGEM Lib、DIASECS Lib 及 SECS SML 共四类。当切换到该种类页面时，可以收到 RunTime 模块在执行时所发出的各类事件。页面提供最近发生的 50 笔事件记录。



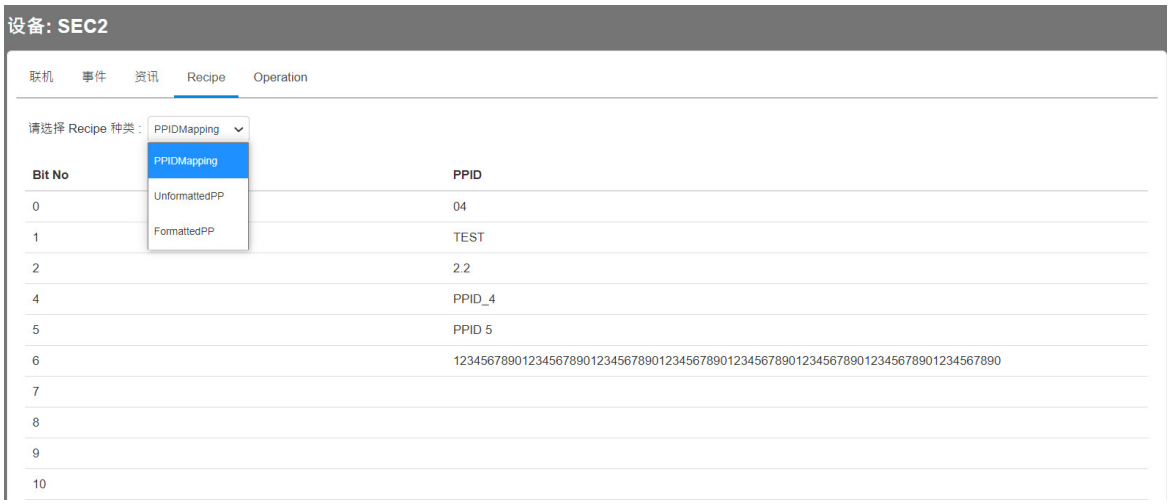
G.3.5 资讯

「资讯」种类分为: Status Variable Information、Data Variable Information 及 Equipment Constant Information 共三类。这三类资讯是用户在配置文件设定并储存于 PLC。欲查看这三类资讯，必须与 PLC 联机后方能查看信息记录 核对 Value 时 需确认 Address 地址是否正确 针对 SV Variable Information、Data Variable Information，若于 SV DATA、DV DATA 设定 Format 为 Link，且无绑定的情况下，于资讯页面的 Format 显示 LIST 且 Length 为 0；若绑定 1 笔数据，Format 将显示被绑定的 Format 且 Length 为被绑定的 Length；若绑定 2 个以上，Format 显示 LIST，而 Length 则依据被绑定的个数呈现。



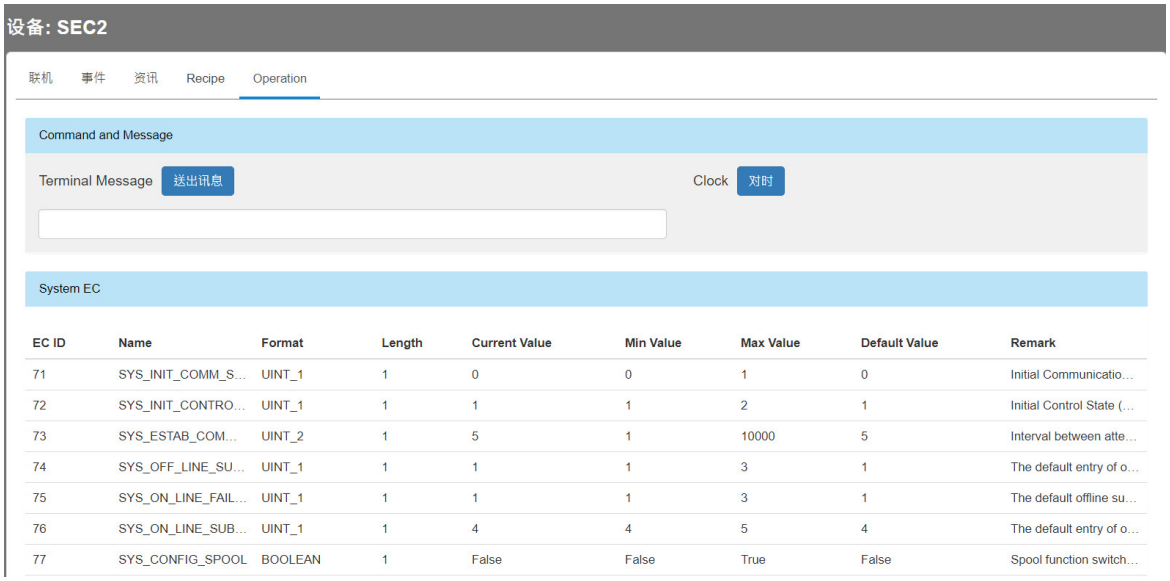
G.3.6 Recipe

「Recipe」种类分为: PPIDMapping、UnformattedPP 及 FormattedPP 共三类。用户在配置文件设定完成且联机后方可在页面查看这三种类型数据。其中 FormattedPP 对照则是由 Formatted Command Code 绑定后的 FormattedPP，若绑定 2 笔以上则会以 2 笔数据显示，若无绑定则不会显示。



G.3.7　Operation

此页面包括送出信息、对时以及 System EC。



- 送出讯息：发送 Terminal Message 命令于 Host，信息可在 HOST 端查看。
- 对时：点击“对时”钮，会从 HOST 收到校正时间，进而更改 Runtime 所在的计算机系统时间。
- System EC：显示 System 设定的 EC 信息。

【注】HOST 的时间必须与 DIALink 计算机的时间一致，否则会影响 Token 时效。

G.3.8　修改 LOG 的进阶设定

建议使用默认的设定，但如果有需要对 LOG 纪录保留时间做调整，请修改（需具备系统管理者权限）C:\DIASECS-GEM RunTime for DIALINK\Config\GemRunTimeConfig.xml 文档中的两个部分：

<item LogFileToZipDate="10" />

<item LogDeleteDate="20" />

其中 LogFileToZipDate 是设定多少天内的 LOG 纪录保持原状，对超过的旧数据进行 ZIP 压缩。而 LogDeleteDate 则是不论是否 ZIP 压缩，超过多少天以上的 LOG 数据都被删除。

另外针对有能力进行除错分析的用户，也提供了修改 LOG 的纪录等级，请修改（需具备系统管理者权限）C:\DIASECS-GEM RunTime for DIALINK\DIASECS-GEM RunTimeForDIALINK.exe.config，其中的 levels 可增加 Info 及 Trace 等更详细的 LOG 信息，但请注意开的信息越详细，占用的磁盘空间越大，建议除错后立即恢复原设定。

```
<rules>
  <logger name="System" levels="Error,Warn" writeTo="WriteTo" />
  <logger name="System" levels="" writeTo="WriteToInfo" final="true" />
  <logger name="DIAGEM_Lib" levels="Error,Warn" writeTo="WriteToGEM_Lib" />
  <logger name="DIAGEM_Lib" levels="" writeTo="WriteToInfoGEM_Lib" final="true" />
  <logger name="DIASECS_Lib" levels="Error,Warn" writeTo="WriteTo" />
  <logger name="DIASECS_Lib" levels="" writeTo="WriteToInfo" final="true" />
  <logger name="SECS_SML" levels="Error,Warn" writeTo="WriteTo" />
  <logger name="SECS_SML" levels="" writeTo="WriteToInfo" final="true" />
  <logger name="CommonPLC_Lib" levels="Error,Warn" writeTo="WriteTo" />
  <logger name="CommonPLC_Lib" levels="" writeTo="WriteToInfo" final="true" />
  <logger name="MQTT_Lib" levels="Error,Warn" writeTo="WriteTo" />
  <logger name="MQTT_Lib" levels="" writeTo="WriteToInfo" final="true" />
</rules>
```


G.4 错误排除

项目	问题状况	问题排除
1	在操作网页时，尽量避免在等待页面中按 F5，否则可能会造成各种异常，如：出现 Timeout 或无法启动 Runtime 等状况	若发生此现象，请单击设备总览后再进入设定页面，即可恢复正常。
2	联机后，若设定为 SECS-I mode，因没有接实体线路，会导致与 HOST 通讯异常，故 SECS Driver 会显示联机又断线的状态	若发生 Driver 显示联机又断线的情况，请检查联机设定是否有误。
3	操作过程中，出现「授权检查失败，目前数据或功能无法使用」的信息	此行为表示系统检查到授权已不存在，请确认授权的 Dongle Key 是否被拔除，请再次插入 Dongle Key 即可正常运作。
4	当 SECS 设备联机时，若于「联机」、「事件」、「信息」、「Recipe」及「Operation」页面拔除 Dongle Key，此时，「联机」页面会呈现联机的状态，而「事件」、「信息」、「Recipe」及「Operation」页面则无任何结果显示	请再次插入 Dongle Key，以确保 SECS 设备能正常运作。
5	执行安装软件时，选定安装项目后，等待时间超过 30 秒以上，系统无任何响应	在开始安装之前可能需要等待 30 秒～2 分钟，当时为正在处理关闭服务、检查文件等行为，请耐心等待。
6	SECS/GEM 模块的 LOG 文件存放于何处?	默认 C：\GEMRunTimeLog 建立每日 log 信息，10 天内的数据会以文件夹存放，超过 10 天则每天压缩成一个 ZIP 档，超过 20 天 ZIP 档就会被删除。

附录H 全域授权与 HTTPS 设置

目录

H.1 设定说明 H-2
H.1.1 启用全域授权 H-2
H.1.2 启用 HTTPS H-5

H.1 设定说明

本系统主要应用于 Intranet，若客户在 internet 环境下使用，则需要考虑数据加密安全性问题，在 DIALink UI 可设定页面开启【全域授权】及设定支持 HTTPS。

- 全域授权：系统提供的所有 Web API 需通过账号、密码取得授权后才能进行访问。
- 手动 HTTPS 设定：启用 HTTPS 协议并使用凭证颁发机构签发的合法有效的数字证书。

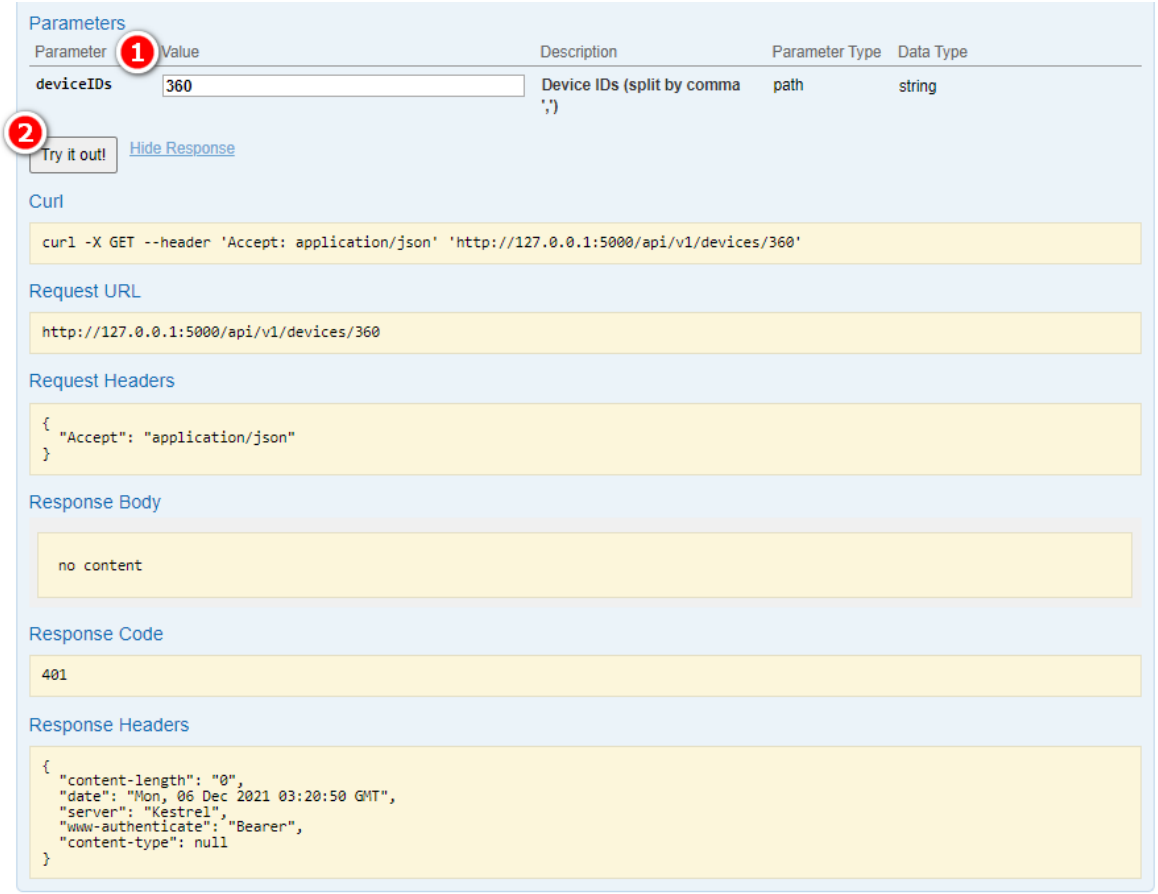
H.1.1 启用全域授权

Step 1：登入系统后至【系统设定】页【一般】启用全域授权后，按下保存变更按钮。

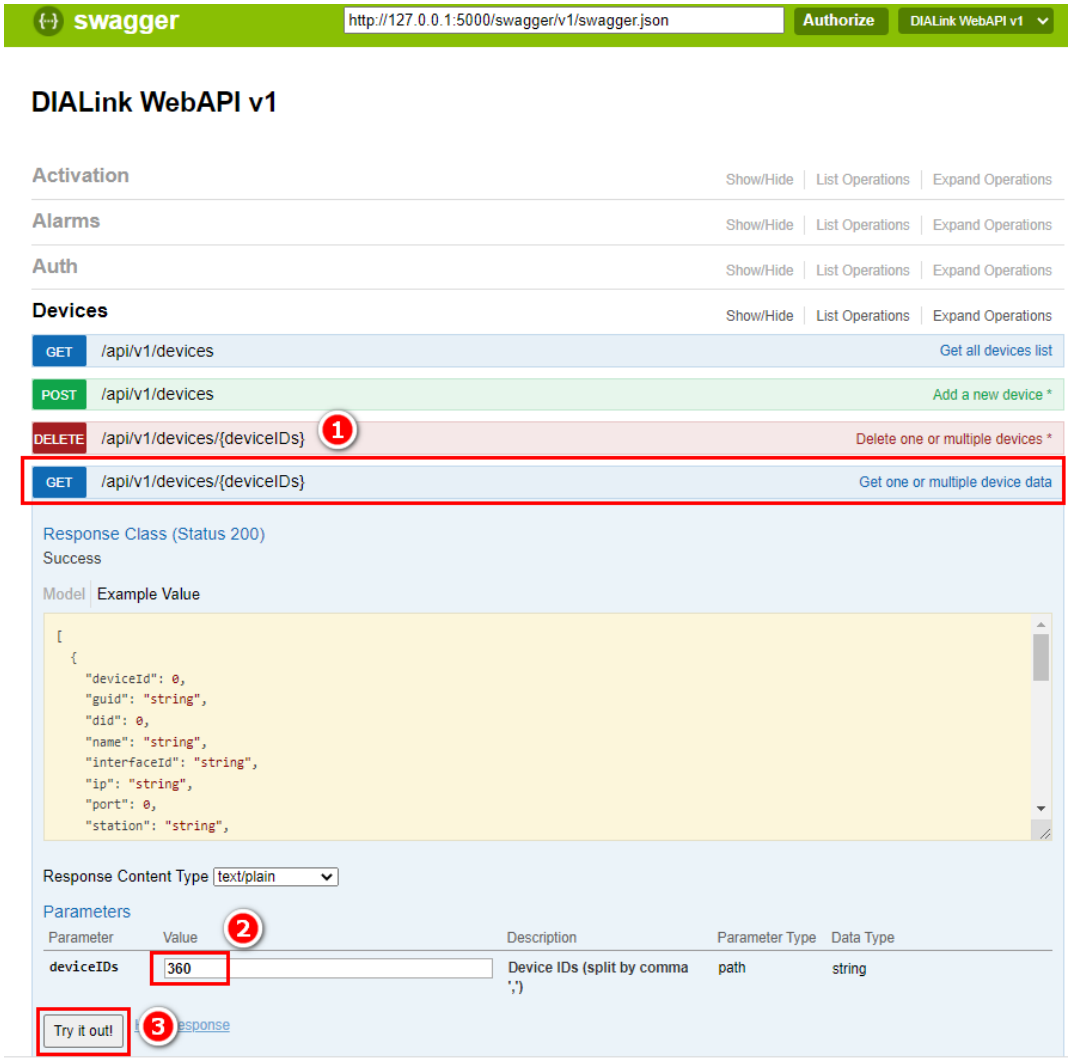


Step 2：开启 Web API Swagger：通过呼叫确认授权机制是否成功运作，我们以 GET /api/v1/devices/{deviceIDs} 来作测试，这是一个 GET 的 Web API， 请输入一个存在 DIALink 中的设备 ID，然后点击 Try it out 按钮。

Step 3：确认【全域授权】：若看到如下图的 Response code：401 的信息就表示确认机制已经开启。



Step 4：改为不启用【全域授权】，按下保存变更，在 Web API Swagger 页面再次以 GET /api/v1/devices/{deviceIDs} 来做测试。输入一个存在 DIALink 中的设备 ID 后点击 Try it out 按钮，在没有开启全域授权机制下，GET 指令可以直接叫用，会回传设备的详细信息。


swagger
http://127.0.0.1:5000/swagger/v1/swagger.json
Authorize
DIALink WebAPI v1
DIALink WebAPI v1
Activation
Show/Hide | List Operations | Expand Operations
Alarms
Show/Hide | List Operations | Expand Operations
Auth
Show/Hide | List Operations | Expand Operations
Devices
Show/Hide | List Operations | Expand Operations
GET /api/v1/devices Get all devices list
POST /api/v1/devices Add a new device *
DELETE /api/v1/devices/{deviceIDs} Delete one or multiple devices *
GET /api/v1/devices/{deviceIDs} Get one or multiple device data
1
Response Class (Status 200)
Success
Model Example Value
[
{
"deviceId": 0,
"guid": "string",
"did": 0,
"name": "string",
"interfaceId": "string",
"ip": "string",
"port": 0,
"station": "string",
Response Content Type text/plain
Parameters
Parameter Value Description Parameter Type Data Type
deviceIDs 360 Device IDs (split by comma ',') path string
2
Try it out!
3

Try it out! Hide Response

Curl

```
curl -X GET --header 'Accept: application/json' 'http://127.0.0.1:5000/api/v1/devices/360'
```

Request URL

```
http://127.0.0.1:5000/api/v1/devices/360
```

Request Headers

```
{
  "Accept": "application/json"
}
```

Response Body

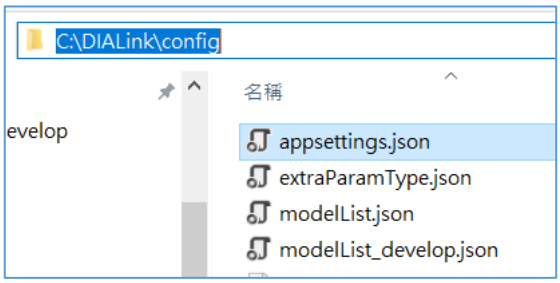

```
[
  {
    "deviceId": 360,
    "guid": "cbbe7470100940d4877c9e37359d6036",
    "did": 11,
    "name": "DVP",
    "interfaceId": "",
    "ip": "192.168.1.66",
    "port": 502,
    "station": 1,
    "discoveryURL": null,
    "serverName": null,
    "comPort": null,
    "baudRate": null,
    "dataBits": null,
    "parity": null,
    "stopBits": null,
    "mode": null,
    "commType": 0,
```


Response Code

```
200
```

H.1.2 启用 HTTPS

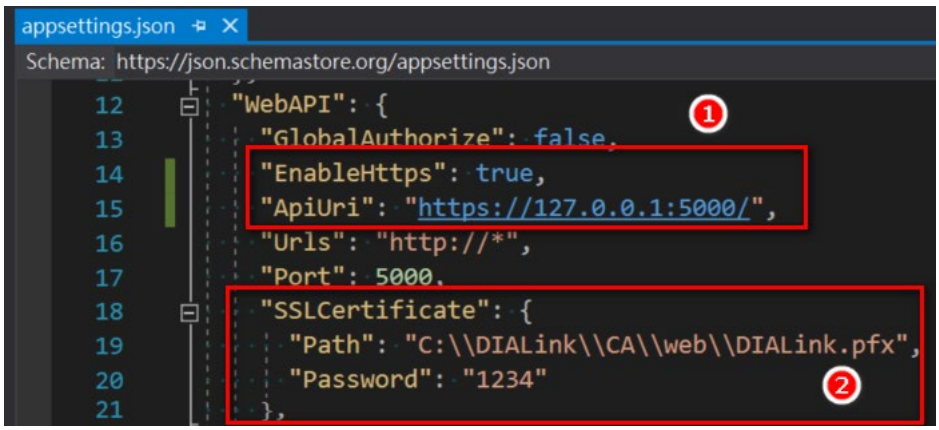
Step 1：至安装路径（默认为 C:\DIALink）下 config 文件夹内，以文本编辑器开启文档 appsettings.json。（需具备管理者权限）



Step 2：在"WebAPI"项目中修改 EnableHttps 属性值为 true，ApiUri 属性值将 http 改为 https。

Step 3：修改 Https 所需凭证文件路径（凭证颁发机构签发的合法有效的数字证书）设定：

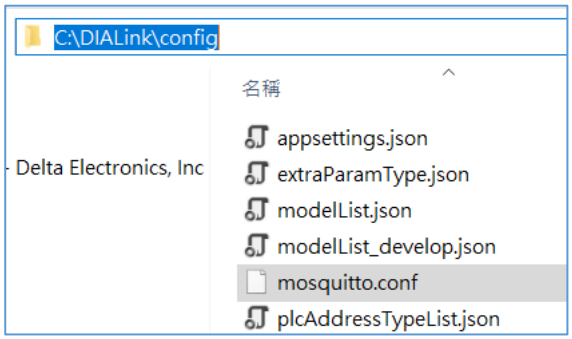
修改 SSLCertificate 区间内分别设定 Path: 用户凭证文档放置路径，Password：凭证密码。此密码在 DIALink 重启服务后会以加密方式保存，译码之后会跟原密码一致。




```
"SSLCertificate": {
  "Path": "C:\\DIALink\\Web\\CA\\web\\DIALink.pfx",
  "Password": "sPapvNQ5tOKBikN6RCwO8w=="
},
```

Step 4：MQTT Web Socket 设定：

至安装路径（默认为 C:\DIALink）下 config 文件夹内，以文本编辑器开启文档 mosquitto.conf（需具备管理者权限）。



Step 5：启用并修改用户凭证文档路径

cafile：凭证文件

certfile：服务器证书文件

keyfile：服务器私钥文件

```
# wss: Web Socket with TLS
# wss listener port-number [ip address/host name]
# CA certificate
cafile C:/DIALink/Web/CA/mqtt/DIALink_CA.crt

# Path to the PEM encoded server certificate.
certfile C:/DIALink/Web/CA/mqtt/DIALink.crt

# Path to the PEM encoded keyfile.
keyfile C:/DIALink/Web/CA/mqtt/DIALink.key
```

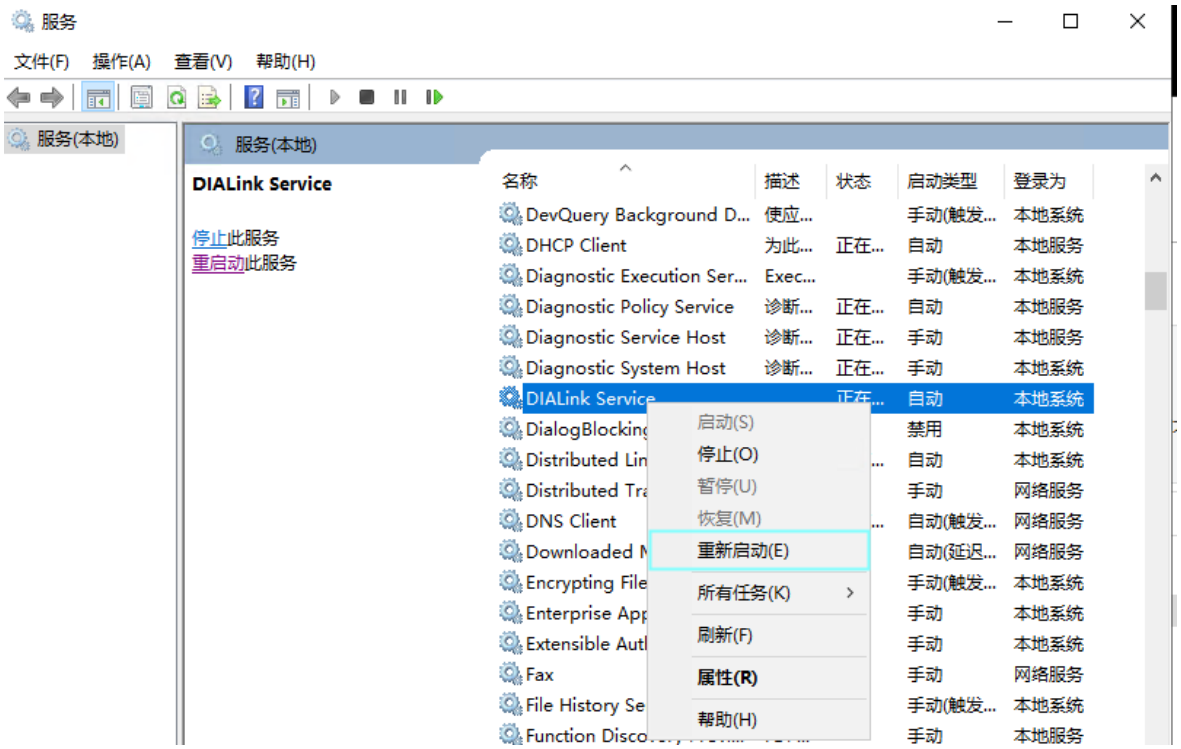
Step 6：至安装路径（默认为 C:\DIALink）下 web\wwwroot\config 文件夹内，以文本编辑器开启文档 config.json（需具备管理者权限）。



Step 7：修改 webSocketEncrypt 属性值设为 true

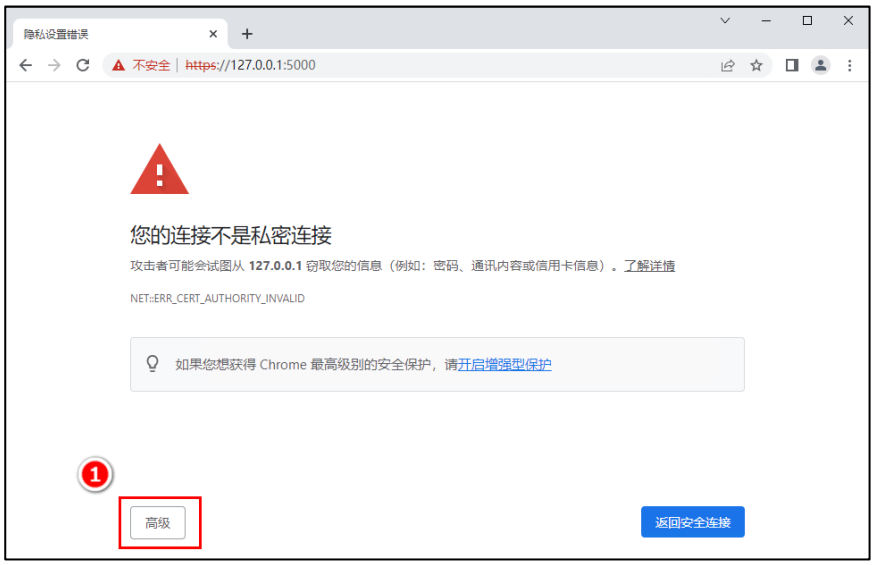
```
{
  "apiUri": "/api/v1/",
  "mqttHost": null,
  "mqttPort": "8081",
  "webSocketEncrypt": true
}
```

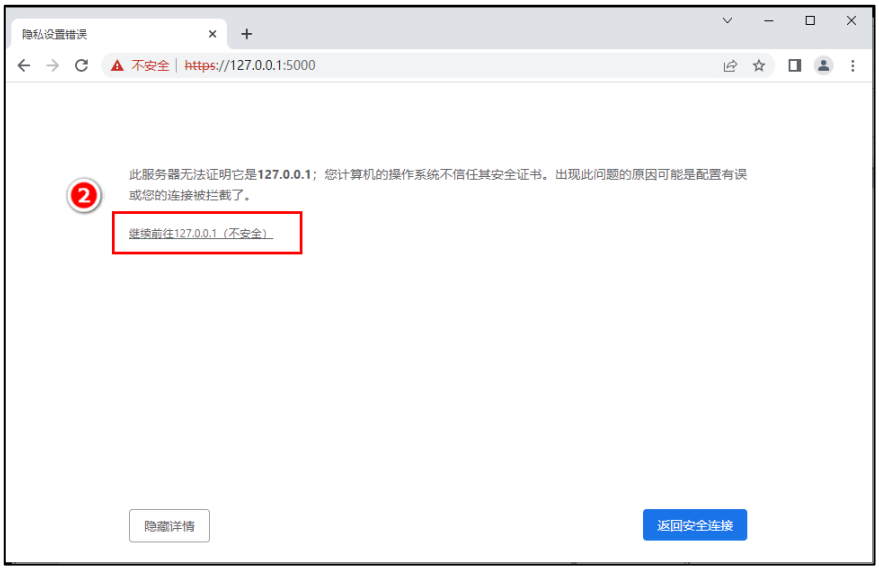
Step 8：重启 DIALink 服务

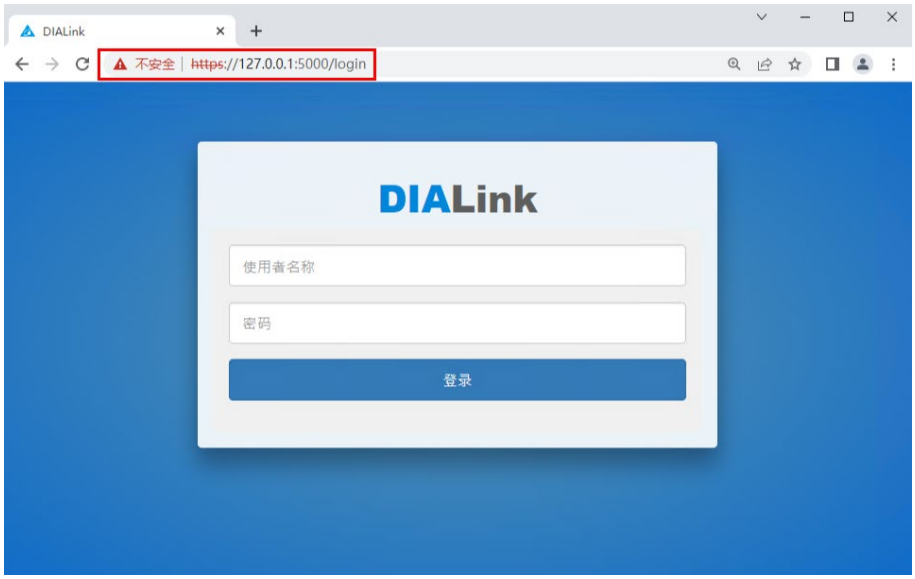


可通过 https://127.0.0.1:Port（此 Port 若使用默认值则为 5000）简易验证网站是否支持 HTTPS 协议。

Note 1：若为不被信任的自签凭证（ex: 系统默认路径下的测试证书文件）则网页一开始会出现以下询问安全性允许的页面，也会影响本系统正常运作 Ex: MQTT Web Socket 通讯、提供第三方呼叫的 Web API。透过【进阶】设定，选择【继续前往网站（不安全）】，则可在浏览器允许下使用 DIALink。




隐私设置错误
不安全 | https://127.0.0.1:5000
2
此服务器无法证明它是127.0.0.1；您计算机的操作系统不信任其安全证书。出现此问题的原因可能是配置有误或您的连接被拦截了。
继续前往127.0.0.1（不安全）
隐藏详情
返回安全连接


DIALink
不安全 | https://127.0.0.1:5000/login
DIALink
使用者名称
密码
登录

MEMO

附录I DIALink 第三方数据串接设置

目录

I.1 第三方数据传输概述 I-3

I.2 Cloud IoT I-4

I.2.1 DIALink 的微软 Azure IoT Hub 连接设置说明 I-6

I.2.1.1 DIALink 与 Azure IoT Hub 数据传输架构 I-6

I.2.1.2 Azure IoT Hub D2C (Device to Cloud) 连接设置说明 I-6

I.2.1.3 C2D (Cloud to Device) 应用说明 I-9

I.2.2 AWS IoT 应用说明 I-12

I.2.2.1 DIALink 与 AWS IoT 数据传输架构 I-13

I.2.2.2 AWS IoT Core 连接设置说明 I-13

I.2.2.3 创建策略 I-14

I.2.2.4 创建物品 I-15

I.2.2.5 证书和密钥 I-18

I.2.2.6 DIALink AWS IoT 设置 I-20

I.2.2.7 AWS IoT 客户端检视 MQTT 信息 I-22

I.3 Webhook I-22

I.4 MQTT I-23

I.5 PC I-24

I.5.1 需求与架构说明 I-24

I.5.2 新增 PC 设备与点位 I-25

I.5.2.1 新增设备 I-25

I.5.2.2 点位设置 I-26

I.5.3 配方信息 I-27

I.5.4 PC 端串接与上传至第三方系统设置 I-28

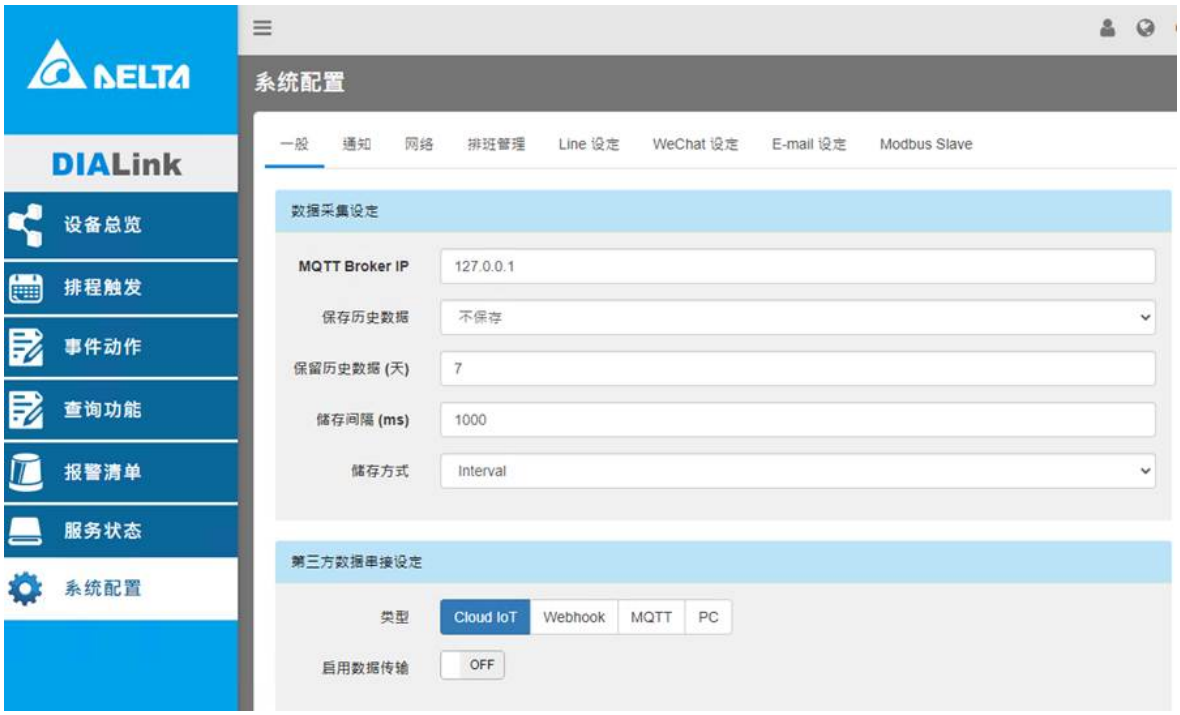
I.5.4.1 PC 端 Http 串行接口设置 I-28

I.5.4.2 上传至第三方系统设置 I-29

I.5.4.3 PC 设备数据上传说明 I-29
I.5.4.4 PC 设备数据上传 HTTP 接口说明..... I-30
I.5.4.5 PC 设备生产配方请求 HTTP 接口说明..... I-31

I.1 第三方数据传输概述

从 DIALink 网页的「服务状态」选单上，可以看到【Data Upload】模块，该模块负责将设备收集上来的数据传送到第三方外部系统。可通过【系统配置】->【第三方数据串接设定】设置开启 / 关闭此功能，目前设定方式有【Cloud IoT】、【Webhook】、【MQTT】以及【PC】。



- 上传数据格式说明

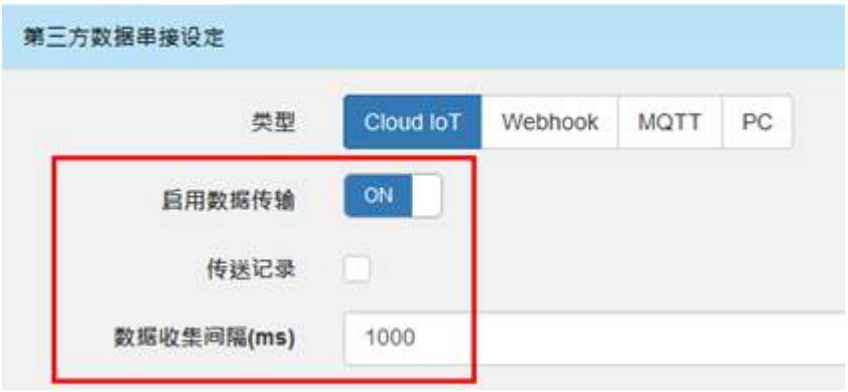
数据传输内容除了【PC】以外都使用以下数据结构，内容是以 JSON 格式呈现的字符串。

参数	数据类型	说明
nodeId	String	节点 ID：每一套 DIALink 唯一标识符
deviceId	Integer	设备 ID
deviceName	String	设备名称
tagId	Integer	点位 ID
tagName	String	点位名称
result	String	点位数值
updateTime	Datetime (UTC)	数值取得时间

```
{
    "nodeId": "cbbe7470100940d4877c9e37359d6036",
    "deviceId": 360,
    "deviceName": "DVP",
    "tagId": 55287,
    "tagName": "1003",
    "result": "55",
    "updateTime": "2021-12-10T04:28:08.3492087Z"
},
{
    "nodeId": "cbbe7470100940d4877c9e37359d6036",
    "deviceId": 360,
    "deviceName": "DVP",
    "tagId": 55289,
    "tagName": "1005",
    "result": "66",
    "updateTime": "2021-12-10T04:28:08.3492087Z"
},
{
    "nodeId": "cbbe7470100940d4877c9e37359d6036",
    "deviceId": 360,
    "deviceName": "DVP",
    "tagId": 55291,
    "tagName": "1007"
    "result": "77",
    "updateTime": "2021-12-10T04:28:08.3492087Z"
}
```


I.2 Cloud IoT

在【第三方数据串接设定】点选【Cloud IoT】页签，点击启用数据传输按钮可开启设置参数项目。目前支持 Azure 及 AWS 云端服务，输入从云端 IoT Hub 得到的信息与收集间隔（ms），即可开始将设置为要上传的点位数据向云端抛出。若勾选【传送记录】，则可在 DIAL_iotHubLog 数据表中查询到上传的记录，每分钟丢出一次，内含这一分钟内所有需上传的点位。依照扫瞄时间将数据上传，假设扫瞄时间设为 20000ms，则每分钟丢出每个点位的信息就会有 3 笔，每一笔的间隔时间为 20 秒，该事务历史记录只保留五天。传送过程若因故无法正确上传云端，在有勾选【传送记录】时，传送失败的记录也会保留在数据库中，当通讯恢复正常时，除了持续将数据上抛，也会将之前传送失败的数据作断线续传。



```
/****** SSMS 中 SelectTopNRows 命令的指令碼  ******/
SELECT TOP (1000) [id]
      ,[data]
      ,[status]
      ,[message]
      ,[updateTime]
  FROM [DIALinkDataCenter].[dbo].[DIAL_iotHubLog]
```

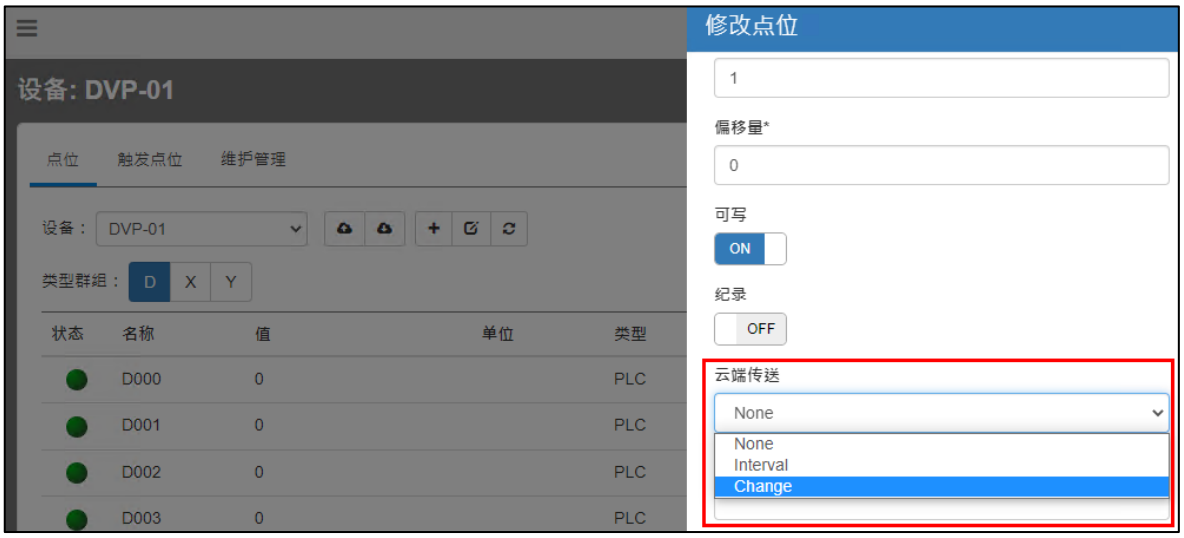
100 %

結果 訊息

	id	data	status	message	updateTime
29	29	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:05:01.327
30	30	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:06:01.337
31	31	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:07:01.347
32	32	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:08:01.360
33	33	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:09:01.363
34	34	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:10:01.380
35	35	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:11:01.383
36	36	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:12:01.383
37	37	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:13:01.393
38	38	[{"nodeId":"cbbe7470100940d4877c9e37359d6036","d...	1		2021-12-10 12:14:01.407

- 点位【**云端传送**】属性：

设备点位的【云端传送】属性可决定该点位是否要在启用数据传输时将此点位的数据传输到云端。若选择【None】，则此点位数据不传到云端。收集频率有两种模式可选择，一种是【Change】模式，此方式仅会在此点位的数据有变更时才会收集点位数据。另一种是【Interval】模式，在间隔时间收集点位数据，每分钟上传云端一次。一个点位的一笔数据大约 0.17Kb，请自行根据预算来决定【云端传送】属性及间隔时间长短。

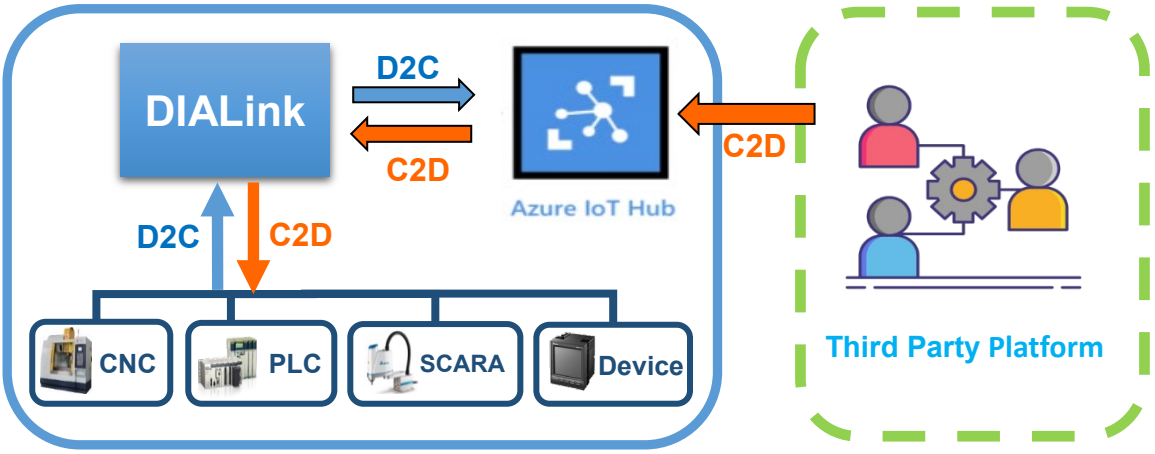


I.2.1 DIALink 的微软 Azure IoT Hub 连接设置说明

进入 Microsoft Azure，再从主页进入 IoT 中心（Hub），可看到 IoT 中心清单（若没有，请创建 IoT 中心），选择要使用的 IoT 中心。



I.2.1.1 DIALink 与 Azure IoT Hub 数据传输架构



DIALink 目前支持设备数据上传至 Azure 及 AWS 云端服务，目前仅支持将 Azure 平台数据写回至通讯设备功能。

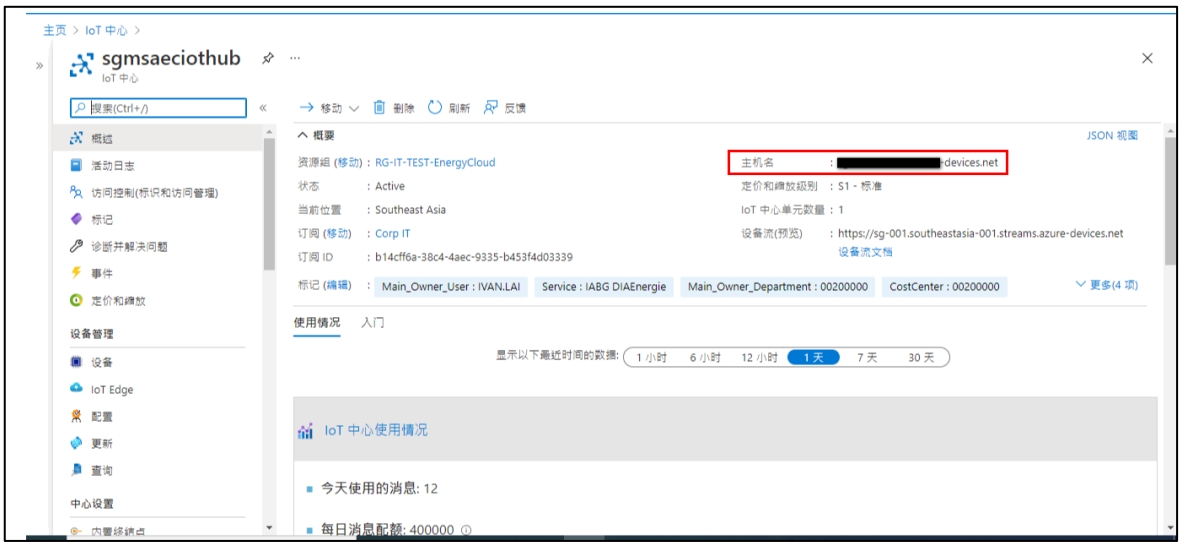
数据传输定义：

- D2C（Device to Cloud）：DIALink 将设备数据传输到第三方云端平台服务接口，目前支持 Azure IoT Hub 及 AWS IoT Hub。
- C2D（Cloud to Device）：用户需通过第三方云端平台提供的 SDK（目前仅支持 Azure IoT 装置 SDK）自行开发程序，将控制命令信息传送到 DIALink，以将数值写入链接的通讯设备。

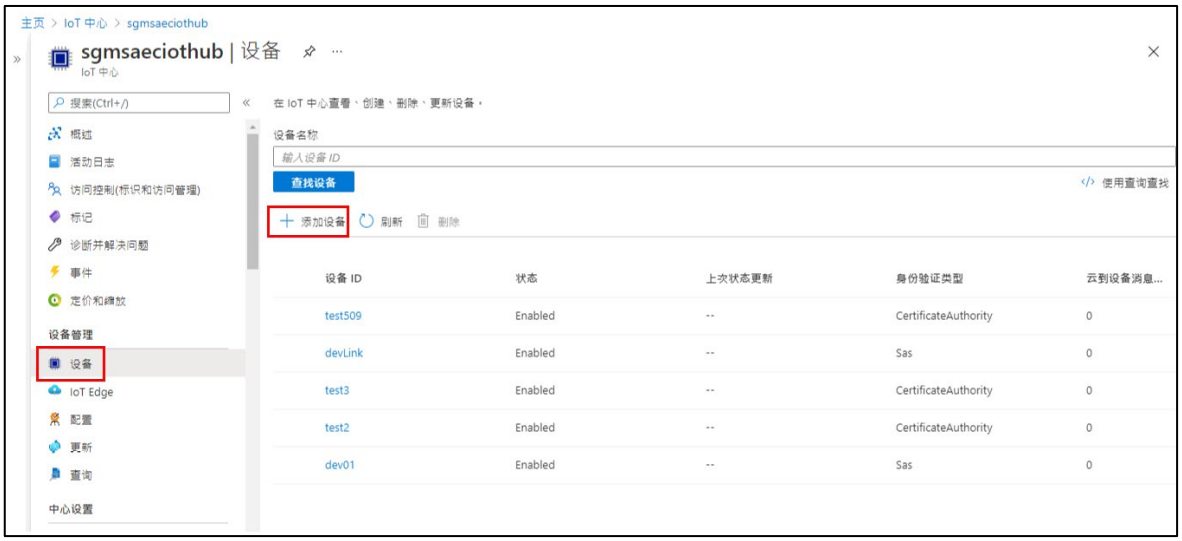
I.2.1.2 Azure IoT Hub D2C（Device to Cloud）连接设置说明

若 DIALink 要将数据传输到云端，仅需在【系统配置】->【第三方数据串接设定】->【Cloud IoT】完成设置并启用即可。

1. 在 IoT 中心画面可看到右边有【主机名】，此名称就是要上传的 URL。

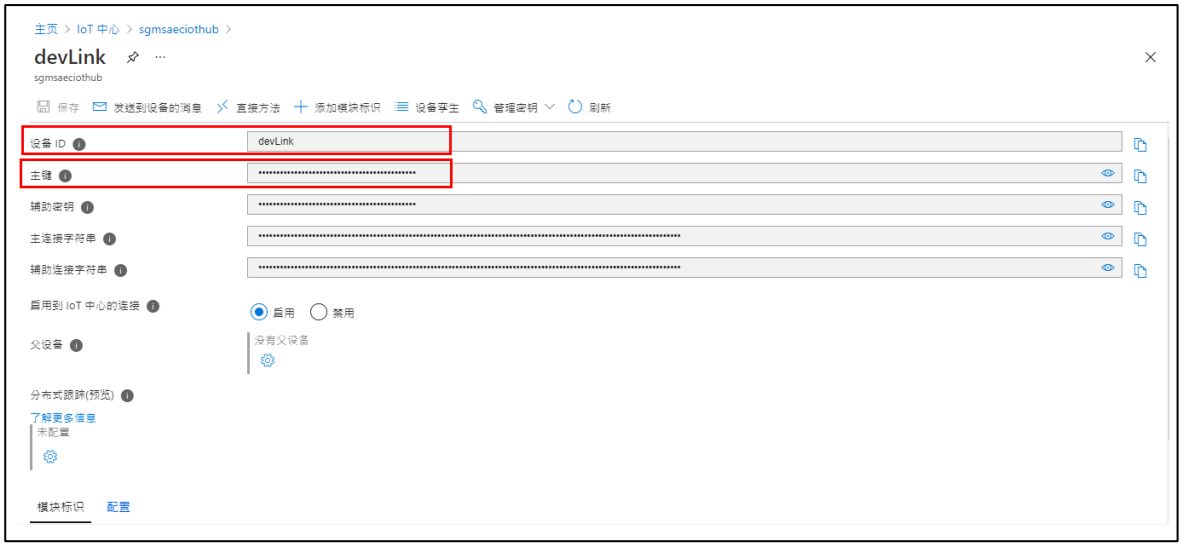


2. 新增设备，有设备才能连接并接收数据，请选择新增设备的验证类型的对应密钥，并启用此设备。





3. 设备的标识符及主要引键则是 DIALink 设定【Cloud IoT】画面中的设备 ID（Device ID）和主键（Primary Key）。



4. 在 DIALink 的【Cloud IoT】设定项目中的【平台】项目点选【Azure IoT Hub】，将上述 Azure 页面上提到的 IoT Hub URL、Device ID 及 Primary Key 信息填入页面字段。



I.2.1.3 C2D（Cloud to Device）应用说明

用户需通过 Azure IoT Device SDK 调用指定的函数名称"DIALinkOnCallMethod"，将命令的内容当作参数，即可对设备进行命令控制。此应用需要更高的云端服务权限，请参考以下说明：

- 在 Azure 平台上 IoT Hub 的**共享访问策略**中，可看到根据权限不同而有着各种等级专属的认证密钥及连接字符串，在 C2D 的程序中我们需要取得【service】权限的连接字符串。



- 以 Azure SDK C#程序为范例如下：

```
using Microsoft.Azure.Devices;
using Newtonsoft.Json;
.....
/// <summary> Control Type </summary>
public enum CloudControlTypeEnum
{
    WriteTag = 0,
}
/// <summary> Control Packet </summary>
public class CloudControlPacket
{
    [JsonProperty("type")]
    public CloudControlTypeEnum Type { get; set; }
    [JsonProperty("data")]
    public string Data { get; set; }
}
/// <summary> Write Packet </summary>
public class CloudWriteTagPacket
{
    [JsonProperty("nodeId")]
    public string NodeId { get; set; }
    [JsonProperty("deviceId")]
    public int DeviceId { get; set; }
    [JsonProperty("deviceName")]
```

```
        public string DeviceName { get; set; }
        [JsonProperty("writeData")]
        public List<CloudWriteTagData> WriteData { get; set; }
}
/// <summary> Write Tag Data</summary>
public class CloudWriteTagData
{
        [JsonProperty("tagId")]
        public int TagId { get; set; }
        [JsonProperty("tagName")]
        public string TagName { get; set; }
        [JsonProperty("value")]
        public string Value { get; set; }
}


// ********************************************************************
// note: the following is a test example
// please fill in the actual data according to your application
// ********************************************************************

// definition of connection variables on the azure iot hub service
string azureIoTHubConnectString = "*****"; // azure iot hub service policy connection string
string azureIoTHubDeviceID = "*****"; //    azure iot hub device id

// specified function name on DIALink and cannot be changed
string deviceCallMethodName = "DIALinkOnCallMethod";

// definition of variables written to the communication device on DIALink system
string nodeId = "ab866d37dc034cdcbbbd7ebc6e5a7e1e";    // the guid on DIALink system
string deviceName = "Delta DVP"; // the communication device name on DIALink system
string tagName = "D"; // the communication tag name of device on DIALink system
int deviceId = 1; // the communication device id on DIALink system
int tagId = 1; // the communication tag id of device on DIALink system

// the writing tag data to communication device
var writeTagList = new List<CloudWriteTagData>();
for (int i = 0 ; i < 10 ; i++)
{
        writeTagList.Add(new CloudWriteTagData
        {
                TagId = tagId + i,
                TagName = tagName + i.ToString(),
                Value = i.ToString()
        });
}

var writeTagData =    new CloudWriteTagPacket
{
        NodeId = nodeId,
        DeviceId = deviceId,
        DeviceName = deviceName,
```

```
    WriteData = writeTagList
};

var cloudControlPacket = new CloudControlPacket
{
    Type = CloudControlTypeEnum.WriteTag,
    Data = JsonConvert.SerializeObject(writeTagData)
};

// create ServiceClient from the specified connection string
ServiceClient serviceClient = ServiceClient.CreateFromConnectionString(azureIoTHubConnectString);

// set response timeout
TimeSpan objTimeSpan = new TimeSpan(0, 0, 0, 5);
// Creates an instance of CloudToDeviceMethod type with response timeout
CloudToDeviceMethod objMethod = new CloudToDeviceMethod(deviceCallMethodName, objTimeSpan);
var controlMessage = JsonConvert.SerializeObject(cloudControlPacket, Formatting.Indented);

// Set payload as json
objMethod.SetPayloadJson(controlMessage);

// interactively invokes a method on DIALink
CloudToDeviceMethodResult result = await serviceClient.InvokeDeviceMethodAsync(azureIoTHubDeviceID,
objMethod);
```

命令信息参数说明

参数	数据类型	说明
type	Integer	控制命令值 0：Write Tag
data	String	命令信息

写入数据(data)参数说明

参数		数据类型	说明
nodeId		String	节点 ID：每一套 DIALink 唯一标识符
deviceId		Integer	设备 ID
deviceName		String	设备名称
writeData		Array	写入数据
	tagId	Integer	点位 ID
	tagName	String	点位名称
	value	String	点位数值

封包范例：

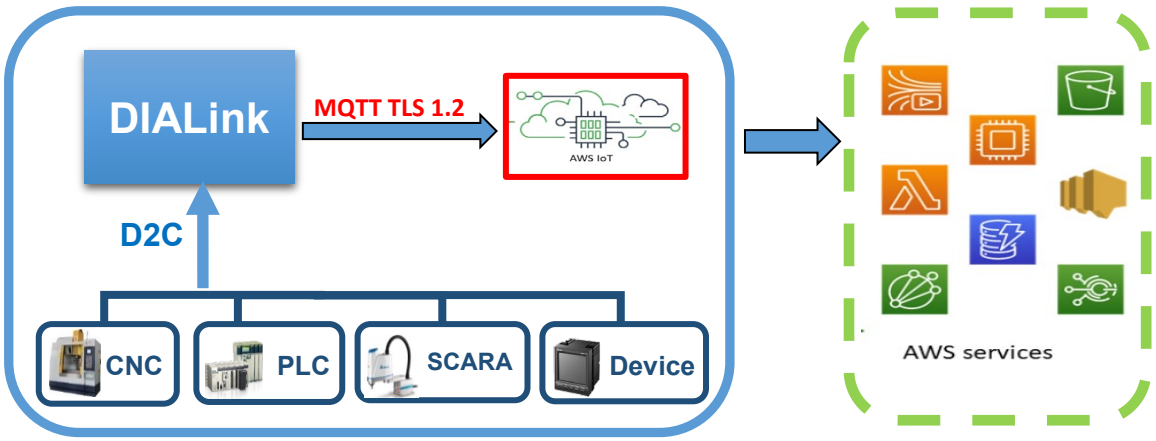
```
{
  "type": 0,                  // Control Type,
  "data": "ControlMessage"    // Control Message
}
```

```
{
  "nodeId": "dc3cb3d3b3b04a219104a125a7983292",
  "deviceId": 1,
  "deviceName": "DeviceName",
  "writeData" : [
    {
      "tagId": 1,
      "tagName": "TagName1",
      "value": "1"
    },
    {
      "tagId": 2,
      "tagName": "TagName2",
      "value": "2"
    },
    {
      "tagId": 3,
      "tagName": "TagName3",
      "value": "3"
    },
    {}
  ]
}
```

I.2.2 AWS IoT 应用说明

AWS IoT Core 提供通过 MQTT 通讯协议将数据内容以 JSON 格式组成并加密上传至 AWS IoT Message Broker。DIALink 作为 IoT Device 角色，仅负责与 AWS IoT Core 连接：客户端网络环境需允许开启对外 8883 端口连接。

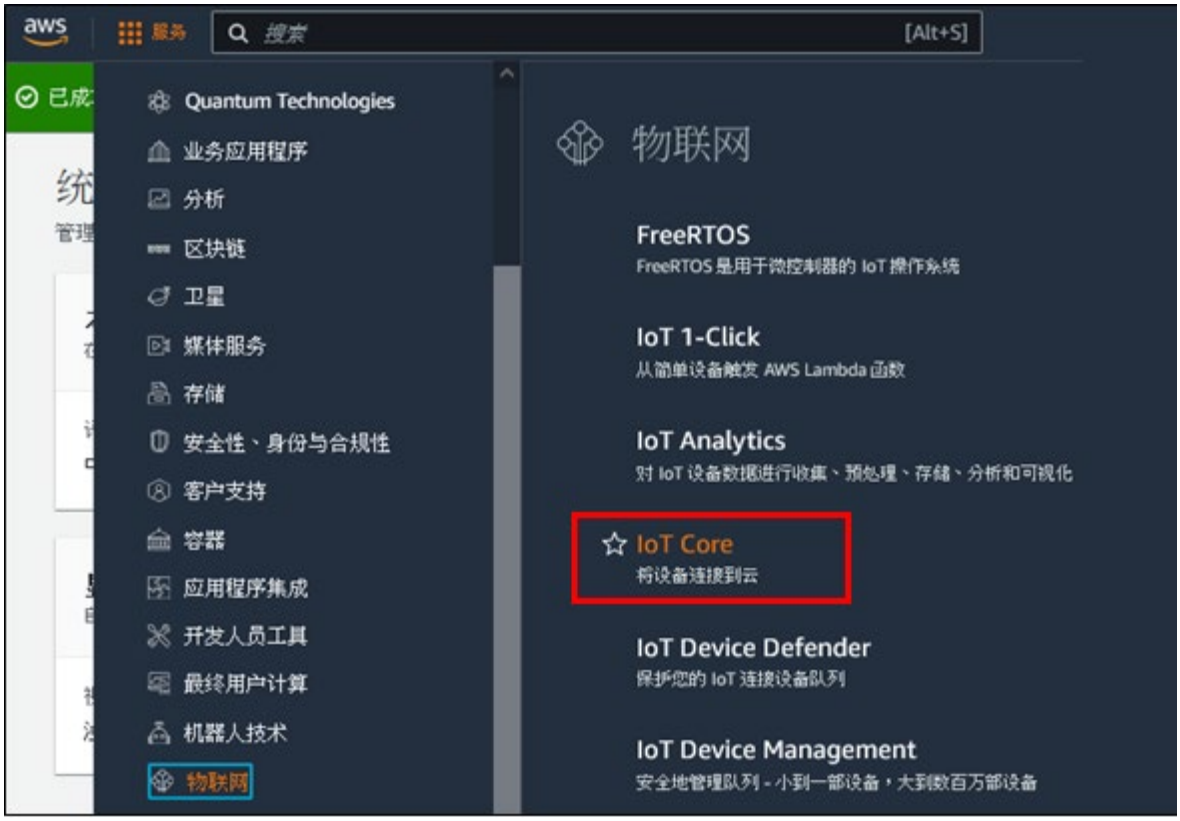
I.2.2.1 DIALink 与 AWS IoT 数据传输架构



（目前不支持 AWS 平台 C2D 数据传输。）

I.2.2.2 AWS IoT Core 连接设置说明

在 AWS 管理控制台网站完成注册后，请登入，从**服务**选择【物联网】，点击进入【IoT Core】。

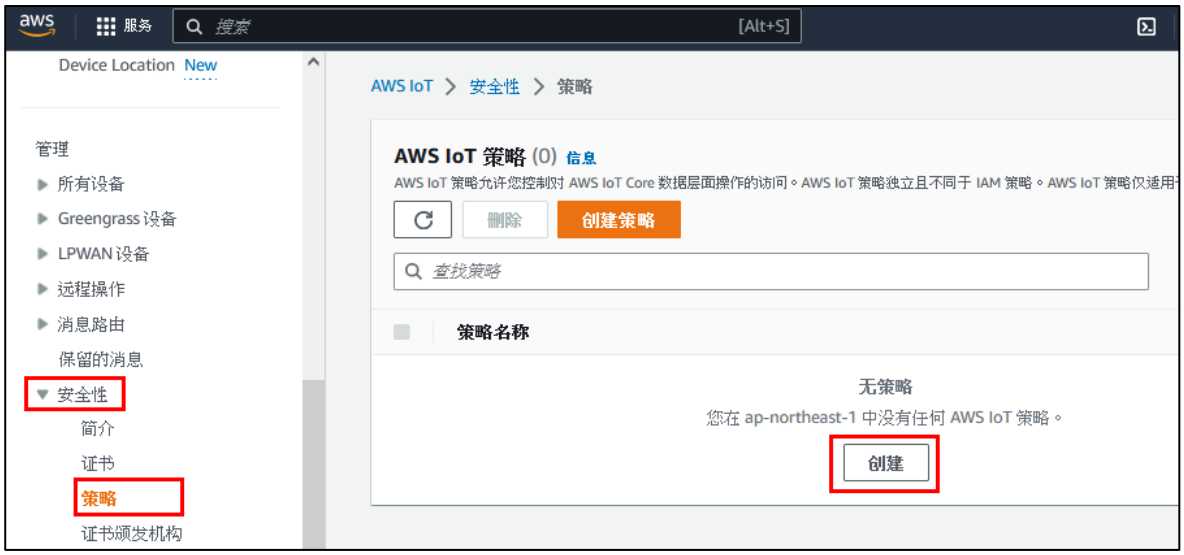


在 AWS IoT Core，收发数据的运作核心是【物品】(Thing)，所以要创建【物品】来将设备数据传输到云端。【创建 IoT Core】，并且下载物品生成证书，此证书需要连接到策略。回到 DIALink 输入 MQTT 相关信息及选择证书，即可完成数据传输。

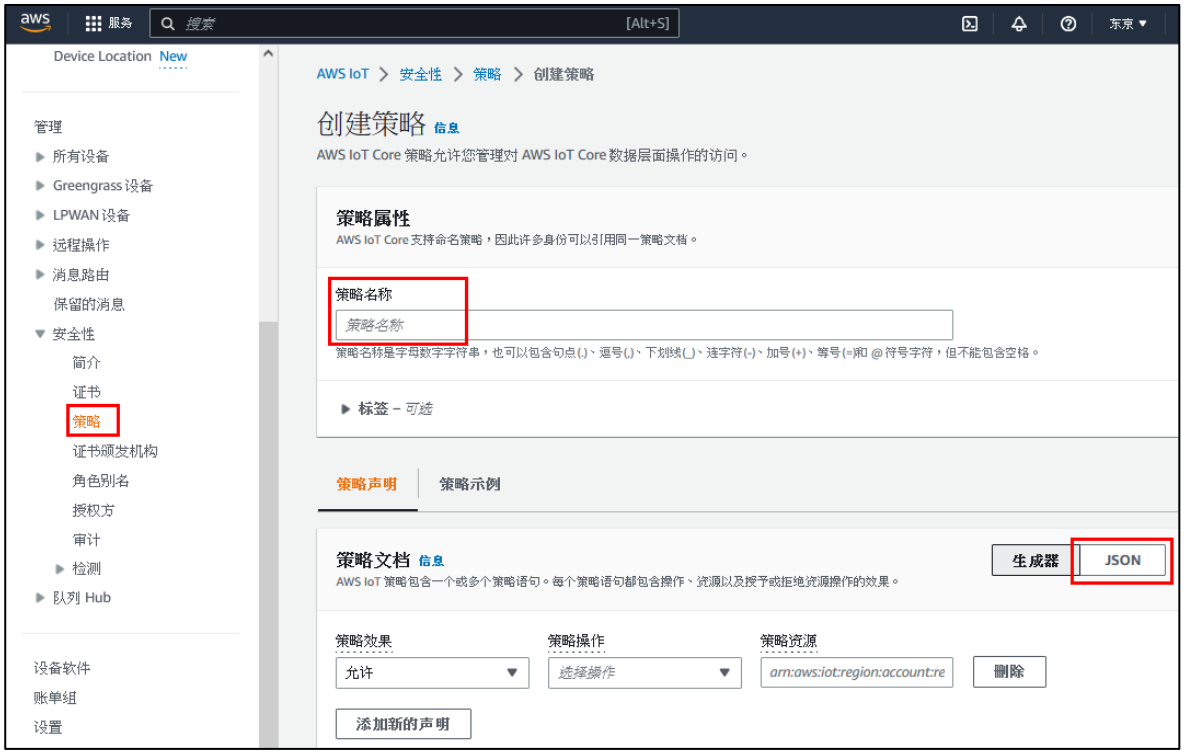
I.2.2.3 创建策略

在开始创建【物品】之前，我们先通过以下 4 个动作创建【策略】，策略可决定数据的访问权限。

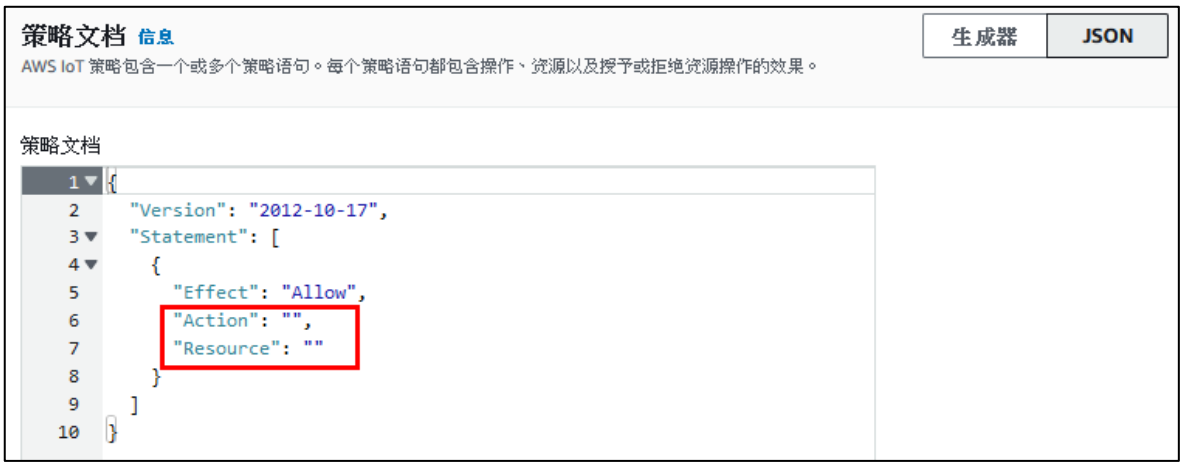
1. 展开左边的【安全性】选项，单击【策略】，在右边的策略画面中单击【创建】按钮。



2. 在【策略名称】属性输入命名，下方右边单击【JSON】按钮。



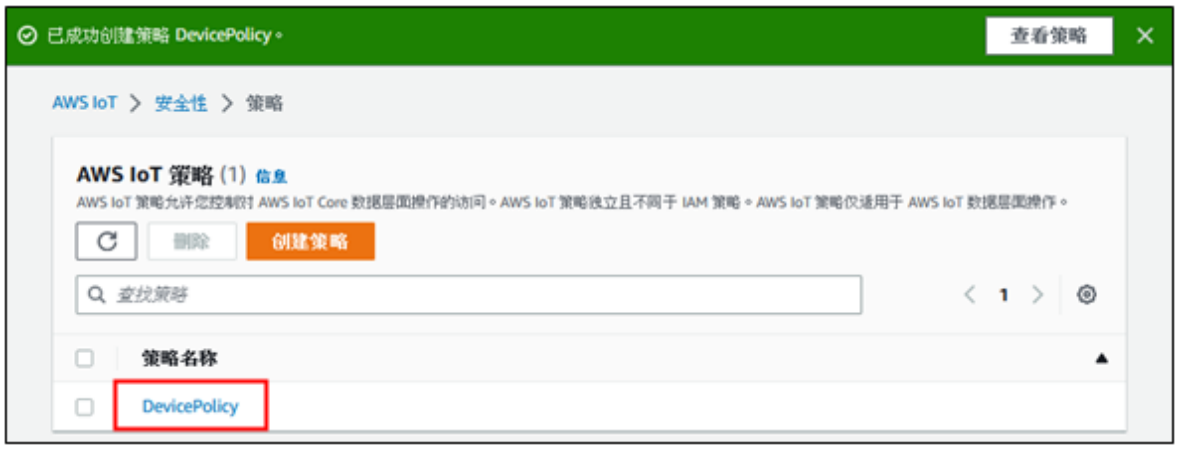
3. 请在【Action】和【Resource】两项输入*，此符号表示所有行动、对所有资源都允许存取。若有信息安全考虑，可限制策略访问权限：例如限制允许的 AWS Region 及 Account Number。



4. 单击【创建】按钮完成创建策略。



完成后可在画面上看到刚刚命名且设置完成的策略。



I.2.2.4 创建物品

开始创建【物品】，通过执行以下五个动作创建【物品】:

1. 在 AWS IoT 左边画面展开【所有设备】单击【物品】，在右边的物品画面单击【创建物品】。



2. 在创建物品画面中单击【创建单个物品】后单击【下一步】。



3. 为此物品命名，输入【物品名称】，在设备影子单击【无阴影】，单击【下一步】。



4. 在设置设备证书画面选择【自动生成新证书】，单击【下一步】。



5. 在附加策略到证书的画面，勾选刚生成的策略，单击【创建物品】按钮。



I.2.2.5 证书和密钥

物品创建完成后，跳出【下载证书和密钥】对话框，提醒用户务必下载设备证书、公有密钥、私有密钥、以及 CA 证书。需特别注意此证书的密钥文件仅在这个画面可以下载，一旦按下【完成】按钮，或鼠标点了非对话框内的其他地方让对话框消失后，就无法再下载密钥文件，仅能重新创建【物品】才能在创建完成的对话框下载到该证书的密钥。



下载后，画面上也有提示用户确实完成下载。下载的策略文件的名称都很长，建议自行变更文件名方便识别。



为了与 AWS IoT 平台创建的 MQTT 连接，需要服务器端 CA 根证书、设备装置证书和私有密钥。DIALink 设备装置证书需采用.pfx 格式，而不是.pem 格式。因此，我们需要通过 OpenSSL 工具将设

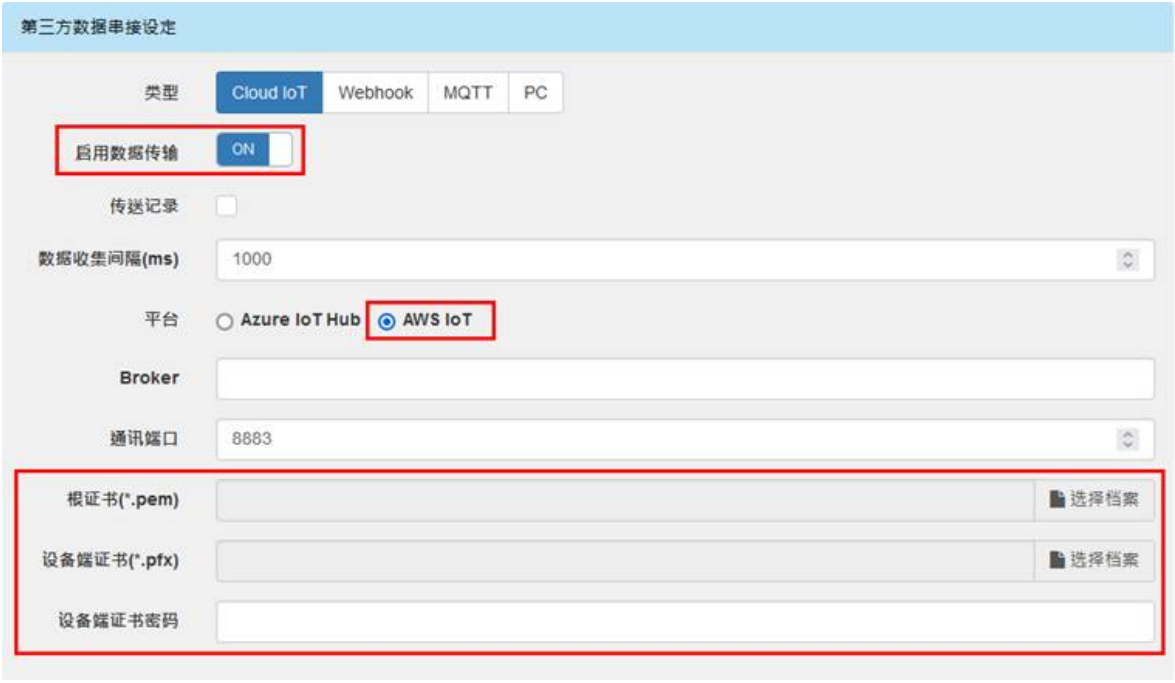
备装置证书从.pem 转换为.pfx，方式如下：

1. 下载安装 OpenSSL 工具软件。
2. 以系统管理员身份开启命令提示字符窗口。
3. 切换到可执行 openssl.exe 的路径，并将刚刚下载的所有证书及密钥文件都移至该目录下。
4. 在命令提示字符窗口执行以下指令，请自行将<>内的文字替换成对应的证书或密钥文件名：
   openssl pkcs12 -export -in <**certificate.pem.crt**> -inkey <**private.pem.key**> -out <**certificate.pfx**> -certfile <**RootCA1.pem**>。
5. 输入指令后，会提示您输入设备装置证书密码，可自行决定是否需要密码。完成后即可得到设备 pfx 格式的证书文件。下例为 AWScertificate.pfx



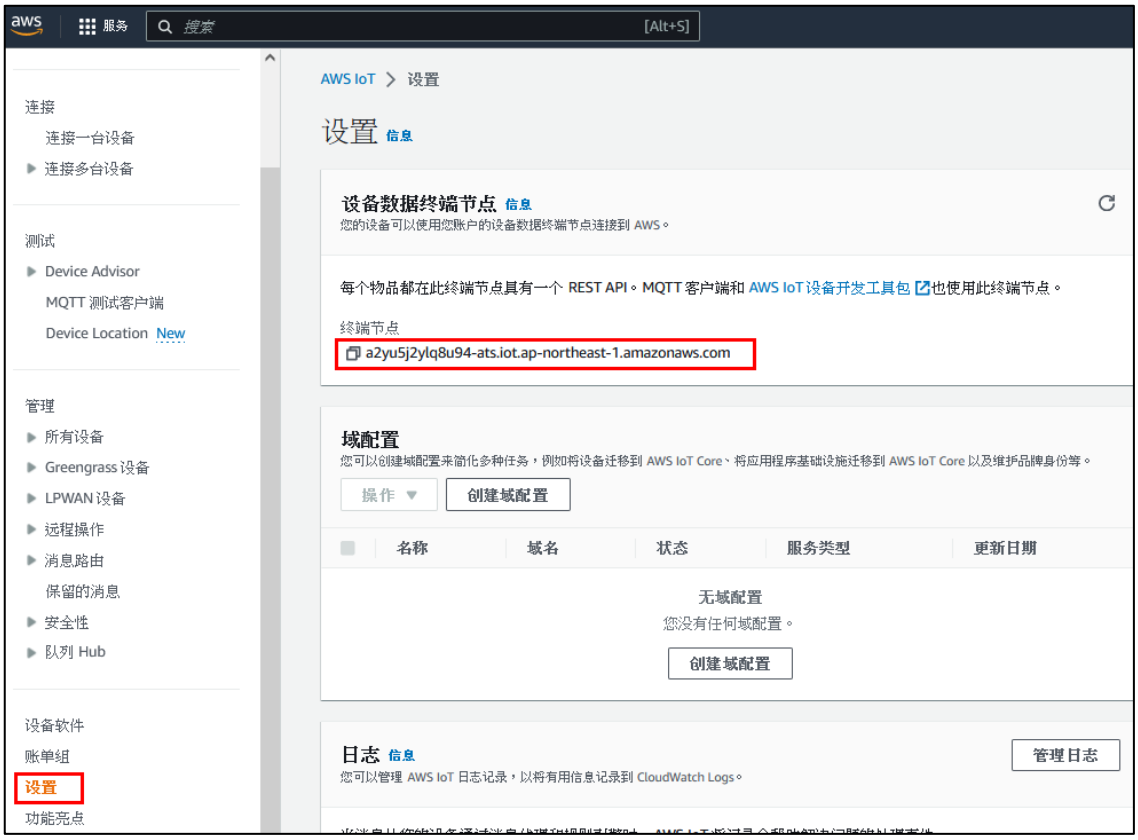
I.2.2.6 DIALink AWS IoT 设置

在 DIALink 的【系统配置】->【第三方数据串接设定】->【Cloud IoT】中，点选【启用数据传输】为 ON，在【平台】项目单击【AWS IoT】，进行两项连接参数设置，变更完毕单击【保存变更】才会生效。



1. 首先需设置的项目为 AWS Message Broker 和通讯端口。

登入 AWS 管理控制台后，进入 AWS IoT 服务画面，单击左边项目列表中的【设置】，即可在右边画面中看到【终端节点】，请复制内容填入 DIALink 画面中 Broker 字段，在通讯端口填入固定值 8883。



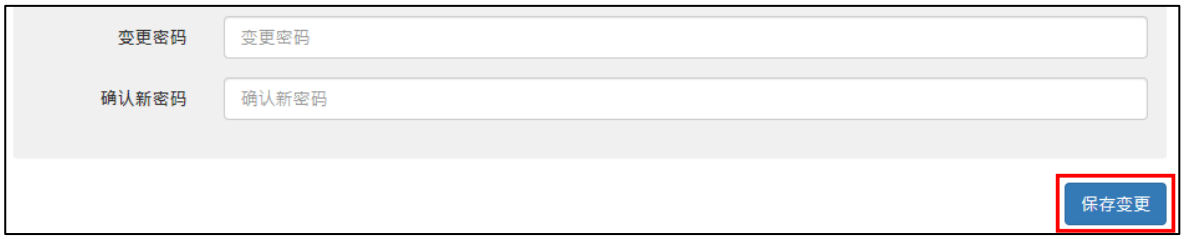


2. 证书文件上传和密码设置

证书文件有 2 个，包含服务器端 CA 根证书，以及上述步骤转出的 pfx 格式设备端证书；如果制作 pfx 设备端证书文件时有给予密码，请在【设备端证书密码】中输入。

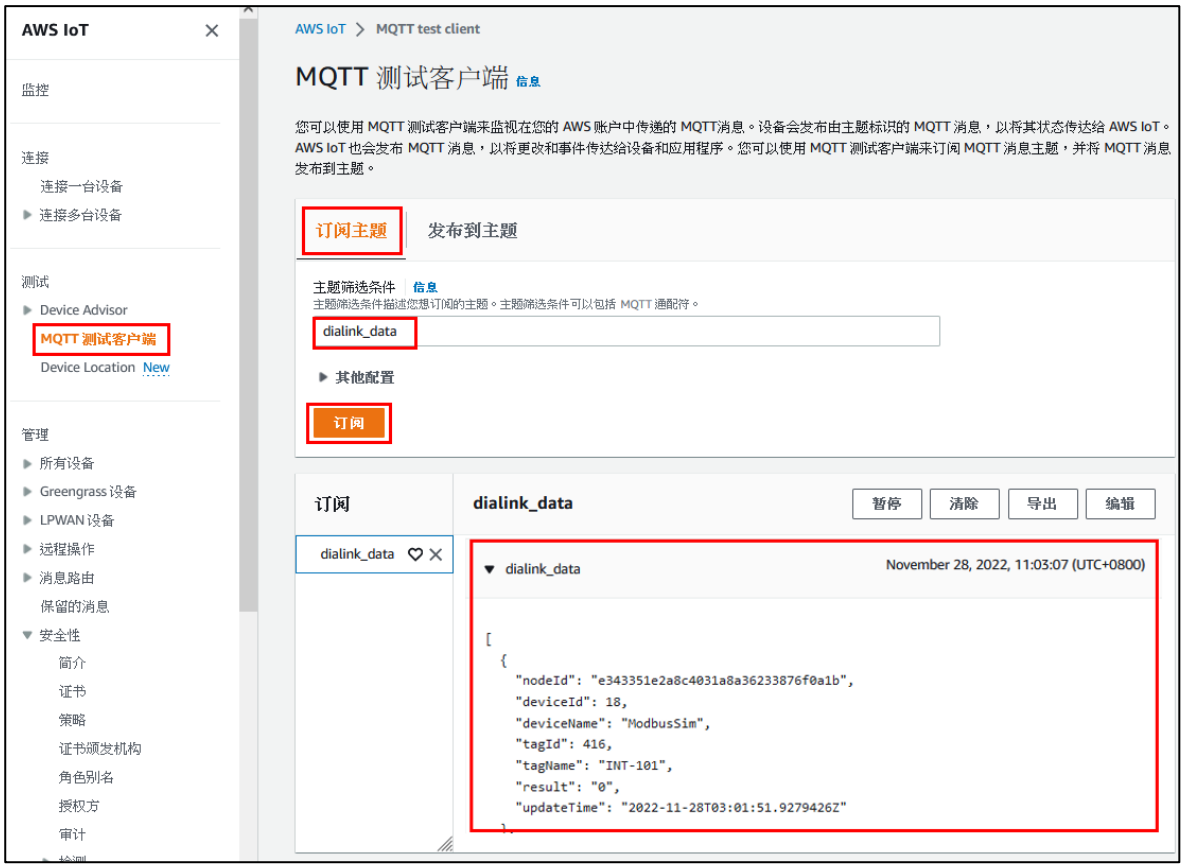


设置完成后请务必按下画面最下方的【保存变更】按钮，结束 AWS IoT 设置。



I.2.2.7 AWS IoT 客户端检视 MQTT 信息

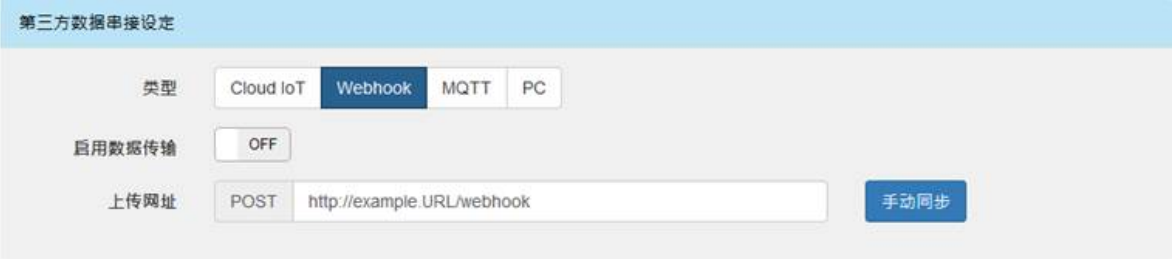
登入 AWS 管理主管台后，在 AWS IoT 画面左边项目中的【MQTT 测试客户端】，可在右边通过【订阅主题】检查 DIALink 是否有将点位数据上传。请在订阅主题输入 dialink_data，若设置都正确，每分钟都能看到有数据上传（至少等待一分钟）。若没有收到数据，请重新检视所有设置步骤，确保所有字段都设置正确，并有启用 AWS IoT 的数据传输。



I.3 Webhook

设置上传网址后将设备更新数据通过 Webhook 以 POST 方式主动上传至第三方程式，变更完毕单击【保存变更】。手动同步则是将所有设备数据一次性手动上传。第三方系统也要有足够的能力实时处理不断被调用的 Web API，以免处理延迟造成数据更新缓慢的假象发生。封包的内容格式请参考 I.1。

由于 DIALink 仅更新有变动数值的点位（Tag），所以外部若需要取得所有点位数据，需在设置并保存上传网址后，单击【手动同步】才会收到多笔数据，每一笔数据包含一个 DIALink 内的所有点位。



I.4 MQTT

设置 MQTT Broker IP（第三方计算机的 IP）、通讯端口、用户名称、密码后，将设备更新数据通过 MQTT 发布（Publish）方式主动上传至第三方 MQTT Broker，变更完毕单击【保存变更】。

同样 MQTT 的内容也是 JSON 格式，说明如下，封包有两个部分，每个封包都带 Topic（主题），其组成为: v1/client/GUID/参数/参数

若要订阅某一设备下所有点位信息，请订阅此 Topic：

范例：v1/client/cbbe7470100940d4877c9e37359d6036/tags/41

项目	参数	说明
1	GUID	DIALink 有一组用于沟通的 GUID，订阅时用来识别是哪个 DIALink 送出的。
2	tagId	此点位的 ID，在单一 DIALink 环境下不重复，多个 DIALink 下则可能重复。
3	Device ID	设备编号。

若要订阅所有设备的所有点位信息，请订阅此 Topic：

v1/client/GUID/tags/+

范例：v1/client/cbbe7470100940d4877c9e37359d6036/tags/+

由于 DIALink 仅更新有变动数值的点位（Tag），所以外部若需要取得所有点位的数据，则需要对该系统内的 MQTT Broker 发布以下 Topic，此为一次性指令，当该系统收到此 Topic，内容也正确的情况下，就会对外将所有点位数据抛出。

v1/client/GUID/rpc/request

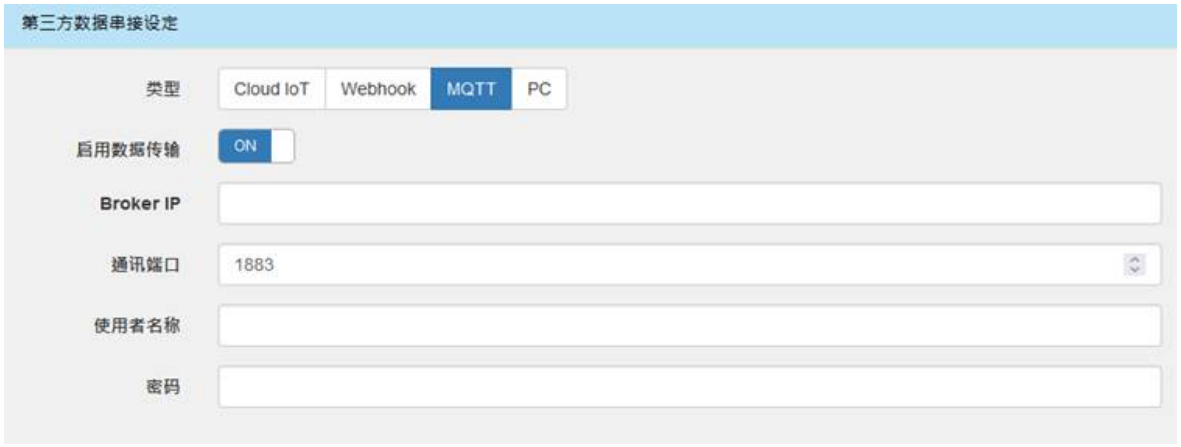
范例：v1/client/cbbe7470100940d4877c9e37359d6036/rpc/request

封包内容

{"type":16}

封包内容也是 JSON 格式

项目	参数	说明
1	type	此处，固定写入 16，系统就会知道此为特殊指令



备注：Tag 数值有变化才会丢出 MQTT 封包，如果某个设备内所有点位（Tag）数值都没有改变，则不会丢出 MQTT 封包。若您连接成功并完成订阅后无法收到任何信息，可将以下两点先厘清：

1. Topic 订阅文字或 GUID 错误。
2. 所有 Tag 数值都没有改变。

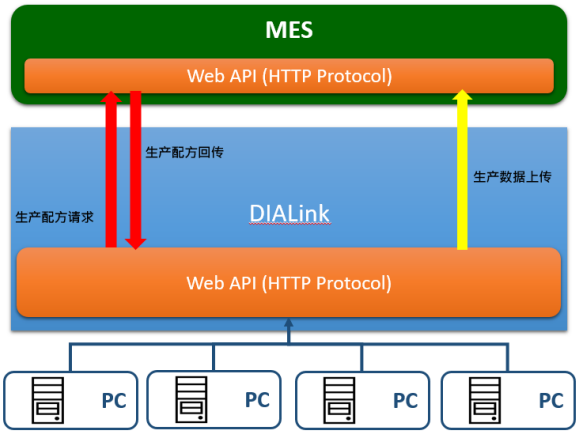
I.5 PC

I.5.1 需求与架构说明

DIALink 支持 PC 设备（Delta 工厂的 PC）数据串接，主要分成两个部份（下图）：

- DIALink 接收设备数据后上传至第三方应用程序(ex: MES…)。
- PC 设备通过 DIALink 向第三方应用程序(ex: MES…)请求生产配方信息。

以上数据更新或最新历史纪录也会呈现在 DIALink 网页上。



DIALink 提供 Web API 作为中间层，内容格式固定。

对下通过 HTTP Protocol 接收 PC 设备：
1. 生产数据内容
2. 生产配方请求

对上通过调用 MES Web API 服务：
1. 将生产数据内容上传
2. 请求生产配方并回传给设备

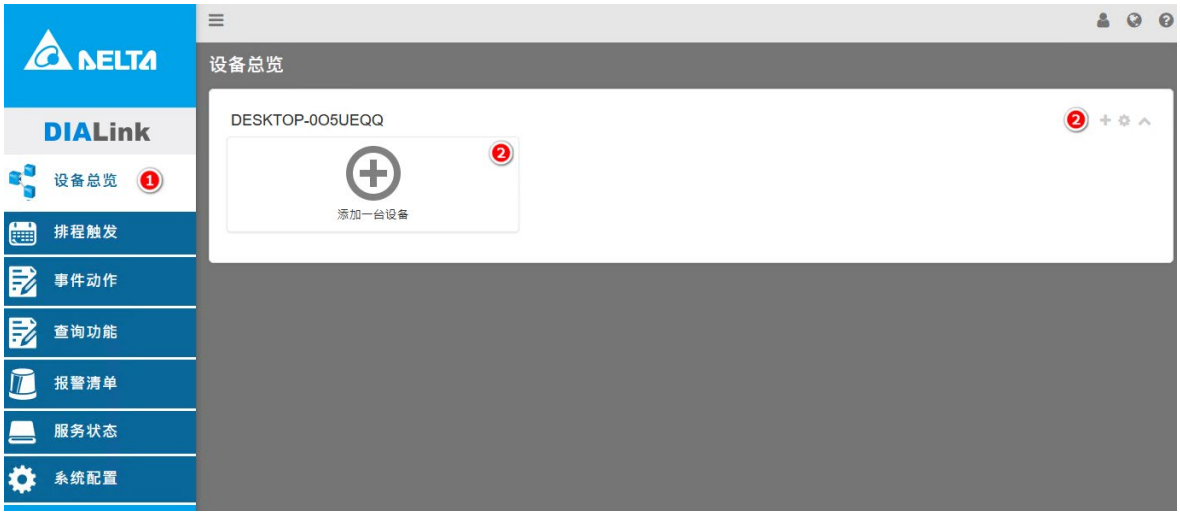
I.5.2 新增 PC 设备与点位

用户成功登入系统后，页面将转到 DIALink 操作界面，系统首页为设备总览，显示本机通过网络 IP 地址相互链接的设备名称，点选 PC 设备区块，可进入该设备看板、设备、点位、触发点位、配方信息功能。(可参考第二章说明)



I.5.2.1 新增设备

依次单击 ❶【设备总览】> ❷【新增一台设备】，若系统上已有设备，单击右上方 ➕ 符号，并在设备类型处选择 PC 即可新增 PC 设备。 (目前 DIALink 仅支持 Delta 工厂 PC 自动设备通过 Http 通讯主动上传特定格式的数据内容)。



- **创建 PC 设备页面**



其中的「Interface ID」字段作为与设备或第三方系统数据串接的唯一标识符，是 PC 设备特有的必填项目。

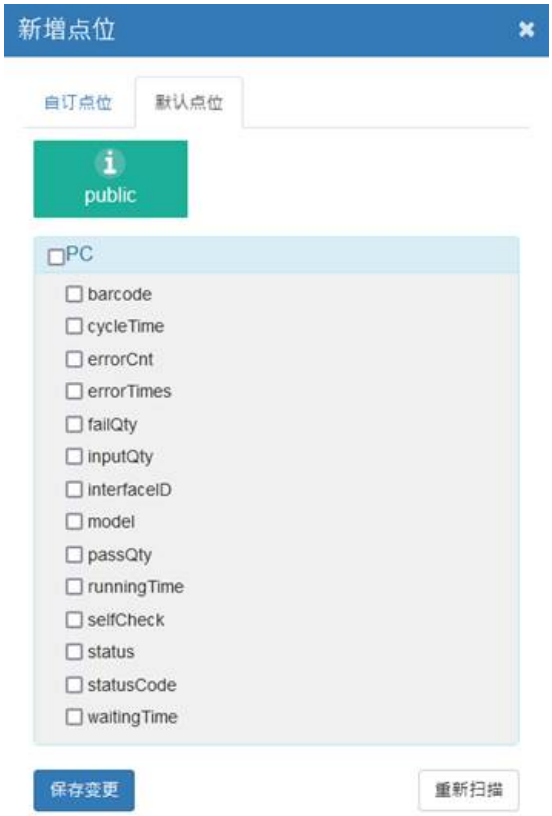
I.5.2.2 点位设置

选择❶【设备总览】后，单击❷或❸设备节点内已新增的设备进入点位设置页面。



类型为 PC 的设备，在新增点位时会先停留在【自定义点位画面】，可切换成【默认点位】分页，用户可根据需求新增 DIALink 内建的 PC 设备默认点位或自行创建特定的点位。点位的新增、编辑、删除部分，请参考第二章的操作说明。

PC 设备默认点位



点位连接：PC 设备点位数据只要上传成功，该点位即会显示连接成功、最新一笔数据以及更新时间。

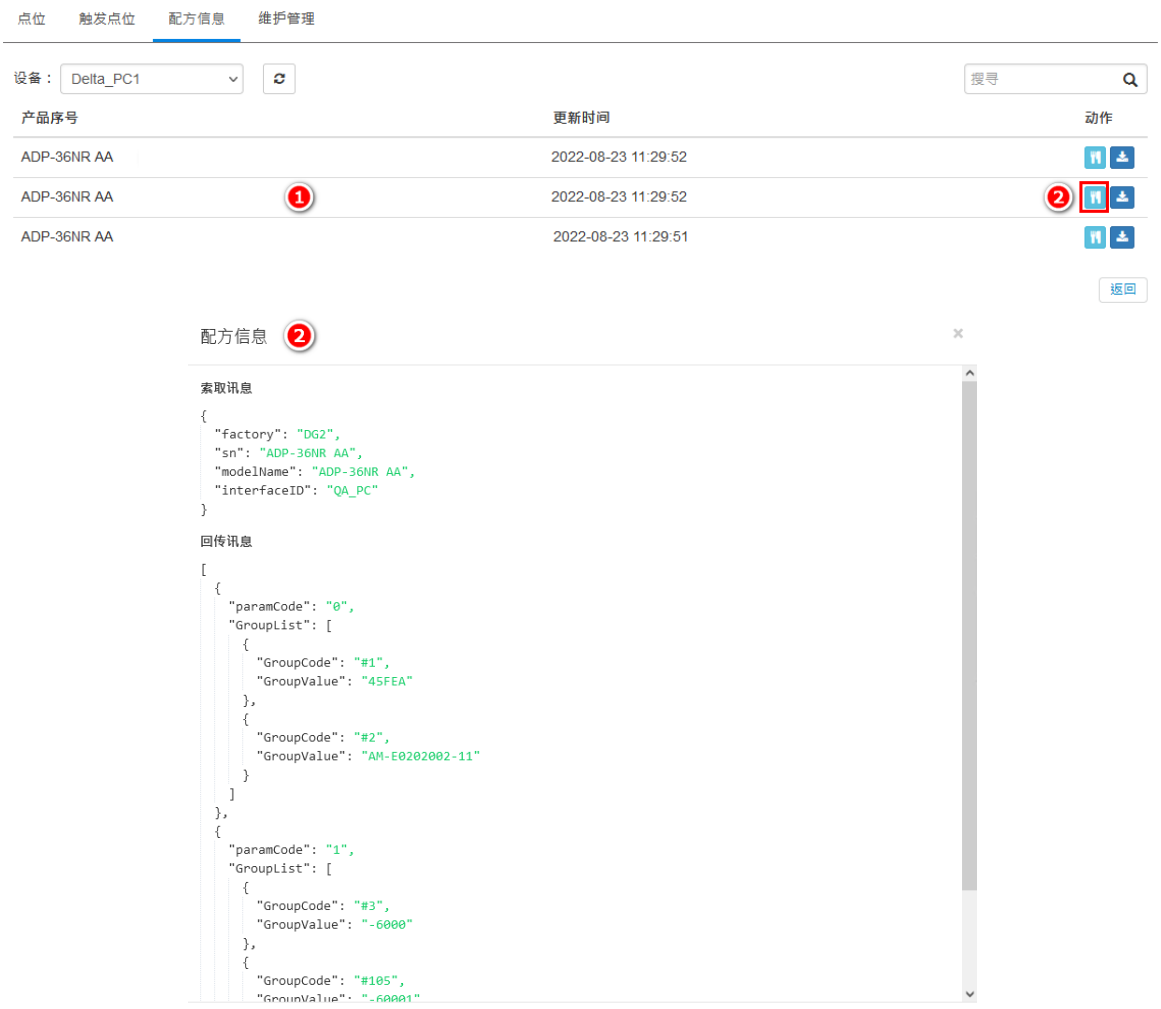
点位断线：PC 设备网络断线，用户在 DIALink 上关闭设备连接。

I.5.3 配方信息

选择❶左方【设备总览】后，单击❷设备节点内已新增的设备进入配方信息页面，PC 类型，会多出【配方信息】页面，在 PC 设备抛出配方信息后，可在此页面看到已发出配方需求的信息。



单击❶或❷可显示该笔记录的配方信息内容。



若要下载该笔设备配方信息内容，❶单击键即可导出。



I.5.4 PC 端串接与上传至第三方系统设置

I.5.4.1 PC 端 Http 串行接口设置

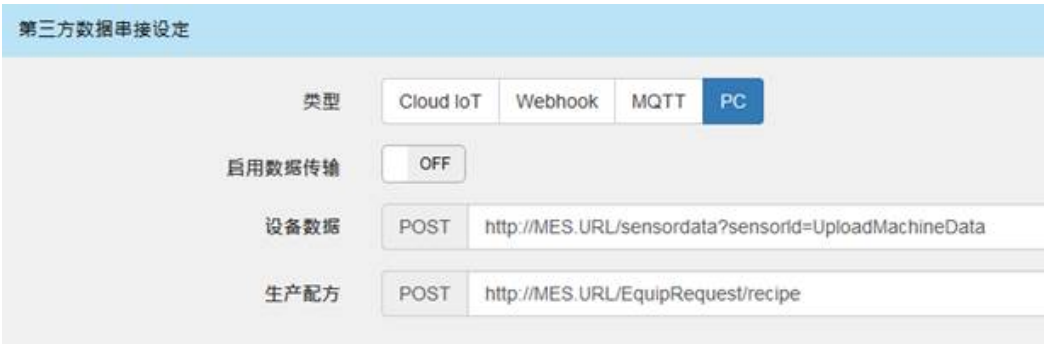
DIALink 系统接收 PC 设备数据上传及配方请求所使用的 HTTP 通讯端口和其他第三方数据串接功能一样，默认为 7638。用户需在 PC 设备设置下面两个 Web API 信息将相关数据或请求传递给 DIALink。

设备数据上传：http://DIALInk IP:7638/sensordata?sensorId=UploadMachineData

生产配方请求：http://DIALInk IP:7638/EquipRequest/recipe

I.5.4.2 上传至第三方系统设置

系统设置画面中的「一般」设置页面里的【第三方数据串接设定】中，单击【PC】类型，设置上传第三方网址，**启用**数据上传功能，将设备更新数据通过 HTTP 以 POST 方式主动上传至第三方程式，变更完毕单击【保存变更】。第三方软件要先了解到 DIALink 搜集的点位有哪些是属于同一设备的，可通过 Interface ID 来识别。



PC 设备串接数据为 JSON 格式，主要分为两种： 设备数据上传、生产配方请求。

I.5.4.3 PC 设备数据上传说明

- 数据分为两部分：公共参数、特殊(差异化)参数。
- 公共参数：每个设备都需统计上传的数据。
- 特殊参数：根据每个设备的不同，用户定义需监测的信息。

参数	含义	规格	备注
interfaceID	设备编号	字符 (50位)	自动化设备的唯一编号
status	设备状态	Integer（1位）	0：正常 1：故障报警中 2：暂停 3：待机（无生产） 4：待料 5：产出满料 6：材料低位预警 7：换线中
statusCode	设备故障时的详细代码	字符（20位）	代码详细内容无需上传，独立提供代码的详细内容给用户
passQty	产出良品数	Integer（8位，10进制）	产品产出

参数	含义	规格	备注
failQty	产出不良数	Integer（8 位，10 进制）	产品不良数
errorCnt	故障次数	Integer（8 位，10 进制）	故障次数
errorTimes	故障时长	Double(10位 ,10进制 ,精确到小数点后2位)	单位：秒 如:381.25
cycleTime	单台产品的生产时间	Double(10位 ,10进制 ,精确到小数点后2位)	单位：秒 产品从进入设备到出去的耗时
runningTime	运行时间	Double(10位 ,10进制 ,精确到小数点后2位)	单位：秒 运行/开机时间
waitingTime	等待时间	Double(10位 ,10进制 ,精确到小数点后2位)	单位：秒 含待机、待料时间
selfCheck	设备自检结果	Integer（1位）	1-OK，2-Fail
inputQty	产品投入数	Integer（8位，10进制）	生产的总数量
barcode	产品序号	字符（100位）	产品的条形码
model	当前机种	字符(25位)	设备当前生产机种
paramList	特殊参数列表		每个设备的特殊监测参数列表
paramCode	特殊参数编号	字符（20位）	设备特殊参数的编号
paramValue	特殊参数值	字符（50位）	特殊参数的值

I.5.4.4 PC 设备数据上传 HTTP 接口说明

- 设备数据通过HTTP协议Restful API将设备数据上传至服务器。
- 上传频率：每生产一台产品/每次设备状态发生变化两种条件都会触发上传。
- 上传数据格式：标准JSON格式，请参考下方的RequestBody。
- interfaceID/设备编号，需可配置。

Description	上传PC设备数据至服务器
API Brief	/sensordata?sensorId=UploadMachineData
HTTP Method	POST
Request Body	[{ "interfaceID":"TEST_EQUIP", "status":1, "statusCode":"ER-02", "passQty":35909,

	"failQty":0, "errorCnt":1, "errorTimes":20, "cycleTime":22, "runningTime":2930, "waitingTime":56.5, "selfCheck":1, "inputQty":35846, "barcode":"NFA", "model":"DPS-NF 03", "paramList":[{"paramCode":"APX0001","paramValue":"17924"}, {"paramCode":"APX0002","paramValue":"17923"}, {"paramCode":"CT_M","paramValue":"1"}, {"paramCode":"CT_Q","paramValue":"0"}] }]
Response Code	Case A: Success processed. Response code:201 Case B: URL incorrect or request data invalid Response code:400 Case C: Specified sensor does not exists Response code:404 Case D: Internal error occurs Response code:500
API URL Sample	http://IP:7638/sensordata?sensorId=UploadMachineData

I.5.4.5 PC 设备生产配方请求 HTTP 接口说明

- 设备生产配方通过HTTP协议Restful API将请求信息上传至服务器。
- 上传数据格式：标准JSON格式，请参考下方的RequestBody。
- 服务器回传生产配方格式： 标准JSON格式，请参考下方的ResponseBody。

Description	设备请求获取配方信息
API Brief	/EquipRequest/recipe
HTTP Method	POST

Request Body	{
	"factory": "DG2",
	"interfaceID": "ADP-36NR AA"
	"sn": "ADP-36NR AA",
	"modelName": "ADP-36NR AA",
	}

Request Parameters Comment	factory：当前厂区信息 interfaceID：设备编号 sn：产品序列号 modelName：当前机种名称 PS: modelName 可为空（默认返回配方）

Response Body	[
	{
	"paramCode": "0",
	"GroupList": [
	{
	"GroupCode": "#1",
	"GroupValue": "45FEA"
	},
	{
	"GroupCode": "#2",
	"GroupValue": "AM-E0202002-11"
	},
	{
	"GroupCode": "#103",
	"GroupValue": "-6000"
	}
	]
	},
	{
	"paramCode": "1",
	"GroupList": [
	{
	"GroupCode": "#3",
	"GroupValue": "-6000"
	},
	{

	"GroupCode": "#105", "GroupValue": "-6000" }] }]
Response Parameters Comment	paramCode：序号 id GroupCode：功能 id paramValue：参数的值 PS: 配方，机种、治具、材料都是配方内容的一部份，都以这种格式一并返回
Response Code	Case A: Success processed. Response code:200 Case B: URL incorrect Response code:404 Case D: Internal error occurs Response code:500
API URL Sample	http://IP:7638/equipRequest/recipe

MEMO