

AH500 快速入门手册

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	2012/06/30
第二版	<ol style="list-style-type: none">1. 更新第 2.3.4 节配线图2. 更新所有软件画面	2012/09/10
第三版	<ol style="list-style-type: none">1. 更新第 2.3.1 节电源模块配线2. 更新第 2.3.4 节模拟输入与输出模块配线3. 更新第 2.6 节的 16AN 数字输出规划地址4. 更新第 2.8 节的 VAR_OUT 为 VAR_OUTPUT	2016/05/05

AH500 快速入门手册

目录

第 1 章 简介

1.1 手册内容简介.....	1-2
1.2 系统架构简介.....	1-3
1.3 主机运作介绍.....	1-10

第 2 章 程序规划编写

2.1 准备工作.....	2-2
2.1.1 硬件.....	2-2
2.1.2 软件.....	2-3
2.1.3 工具与材料.....	2-3
2.2 安装.....	2-4
2.2.1 安装模块.....	2-4
2.2.2 安装脱落式端子.....	2-5
2.3 配线.....	2-7
2.3.1 电源模块配线.....	2-7
2.3.2 数字输入模块配线.....	2-9
2.3.3 数字输出模块配线.....	2-9
2.3.4 模拟输入与输出模块配线.....	2-10
2.3.5 送电.....	2-12
2.4 范例说明.....	2-12
2.5 建立项目.....	2-14
2.6 规划硬件架构.....	2-16
2.7 建立全域符号.....	2-21
2.8 建立功能块.....	2-23
2.9 建立主要程序.....	2-31

第 3 章 程序下载与监视

3.1 COMMGR 设定.....	3-5
3.1.1 启动 COMMGR.....	3-5
3.1.2 开启 COMMGR.....	3-5

3.1.3 设定 COMMGR	3-6
3.2 专案下载	3-7
3.2.1 设定项目通讯	3-7
3.2.2 下载硬件设定	3-8
3.2.3 下载程序内容	3-9
3.3 程序监视与除错	3-10
3.3.1 程序监视	3-10
3.3.2 程序与系统除错	3-15



第1章 简介

目录

1.1 手册内容简介.....	1-2
1.2 系统架构简介.....	1-3
1.3 主机运作介绍.....	1-10

1

1.1 手册内容简介

针对 AH500 PLC 系列的产品，台达依照不同的应用需求，分别为用户准备了不同的说明手册。

AH500 硬件手册：介绍所有硬件的规格、寻址、配线、维护及故障排除等内容。

AH500 操作手册：介绍硬件的配置、联机的设定、主机的运作及软件的设定操作等内容。

AH500 程序手册：介绍主机内装置的定义与所有指令的内容。

AH500 模块手册：介绍各模块的规格、安装、设定及故障排除等内容。

AH500 运动控制模块手册：介绍运动控制模块的规格、配线、指令及功能范例。

ISPSOft 使用手册：介绍编辑软件的详细使用，包含变量、联机、程序及功能块 (FB) 等内容。

PMSOft 使用手册：介绍运动控制模块编辑软件的详细使用，包含编辑模式、联机及加密设定等内容。

AH500 快速入门手册 (本手册)：介绍如何让用户能先快速的建立并使用本系统。本手册除了介绍基本的系统架构外，也利用简单的实例来教导用户，如何一步一步的规划整个系统架构并撰写程序，最后下载并让主机运作，其中包含了变量、功能块 (FB) 等实际应用方式，让用户能快速的体验新功能所带来的便利性。若于系统运行时发生错误信息，请参考第 3.3.2 节 (故障排除的详细说明请参考 AH500 操作手册第 12 章)。

手册中的图示说明

圖示	說明
	点击鼠标左键
	点击鼠标右键
	快点两下鼠标左键
	按住鼠标左键后进行拖曳
	使用键盘输入
	操作顺序 (当说明中需要特别表达操作顺序时会配合此标注，例如  、 )
	配合图片说明的指示编号

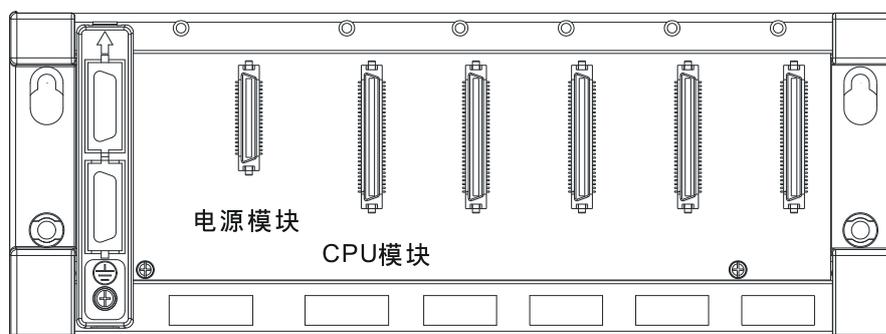
1.2 系统架构简介

AH500 PLC 为台达中型的可程序化控制系统。除了执行速度与储存容量的提升之外，亦支持完整的功能块 (Function Block) 程序开发功能，同时为了符合用户更高端的应用需求，提供了更加灵活的系统扩充架构。在这样的一个系统架构下，用户不会因为系统点数过多或设备距离过远等问题，而必须将系统拆分成多个主机系统来控制，如此一来，可以保留系统的完整性，也让用户在项目的开发过程中，能更加有效率。

AH500 系统最小架构需求：

要架构一个 AH500 系统，至少必须包含一个主背板、一个电源模块、与一个 CPU 模块，这样主机就可以进行规划与运作。

主背板 (4 槽 AHBP04M1-5A)

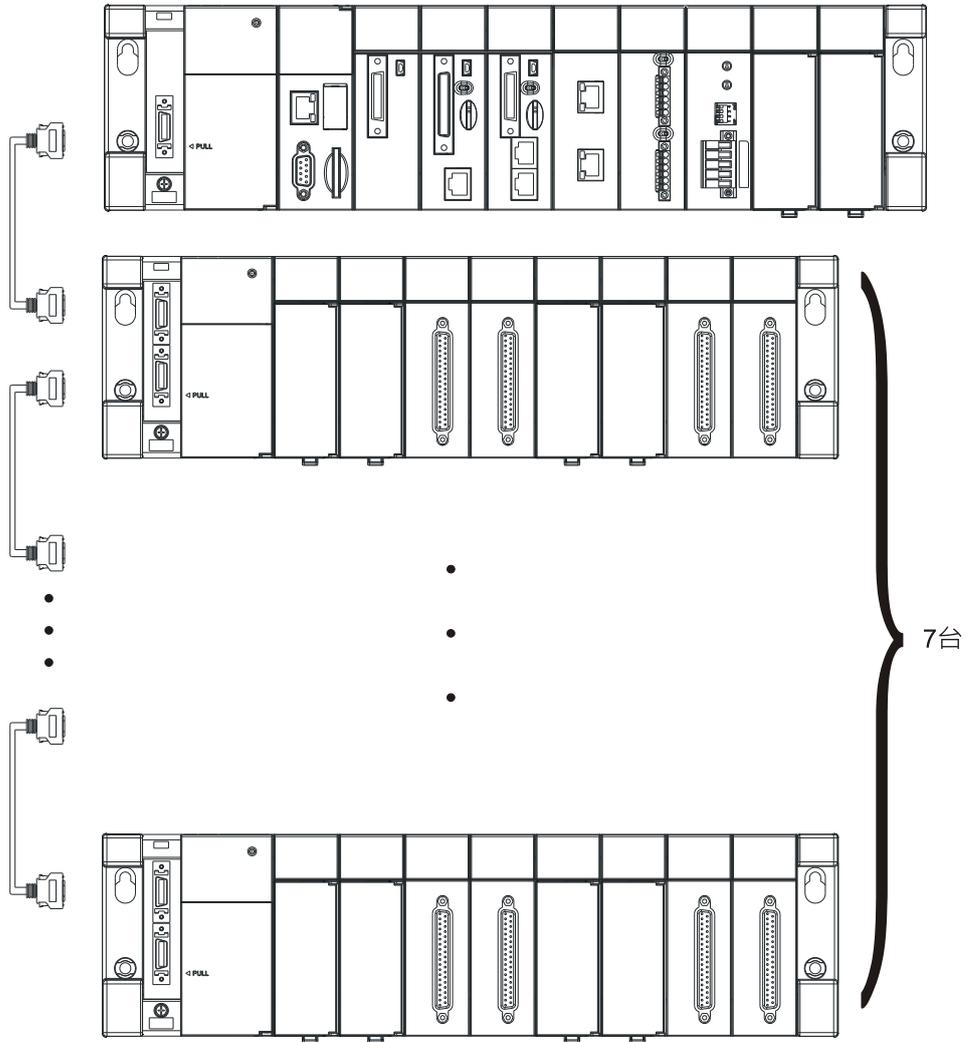


1

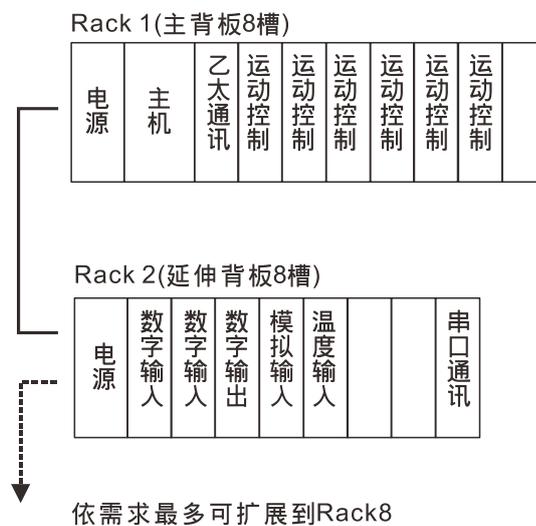
AH500 常用系统架构 (OEM 设备制造业) :

在 OEM (设备制造业) 应用领域里 , AH500 可符合多数的设备开发需求 , 一般来说 , 大多选用一个 8 或 12 槽的主背板即可满足需求 , 而某些高阶设备机种在考虑点数、轴数、布线等因素下 , 也可以在原架构下搭配使用一个 6 或 8 槽的扩充背板 , 以增加系统的点数、轴数或是降低大型设备的布线成本。

架构 : 机架 1 (Rack1) ~ 机架 8 (Rack8)



规划如下



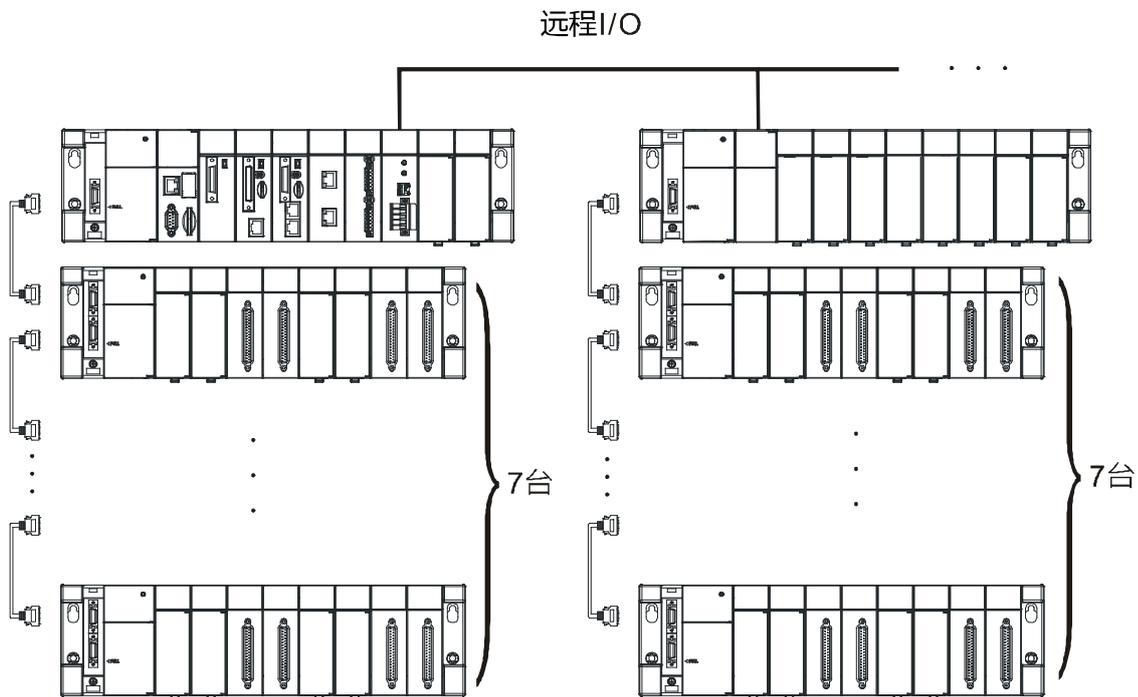
注意：运动控制类型与通讯类型的模块（串口通讯除外），因数据传输速率考虑，安装时必须与主机在同一个背板，否则系统将无法进行规划与运作。

1

AH500 常用系统架构 (SI 系统整合) :

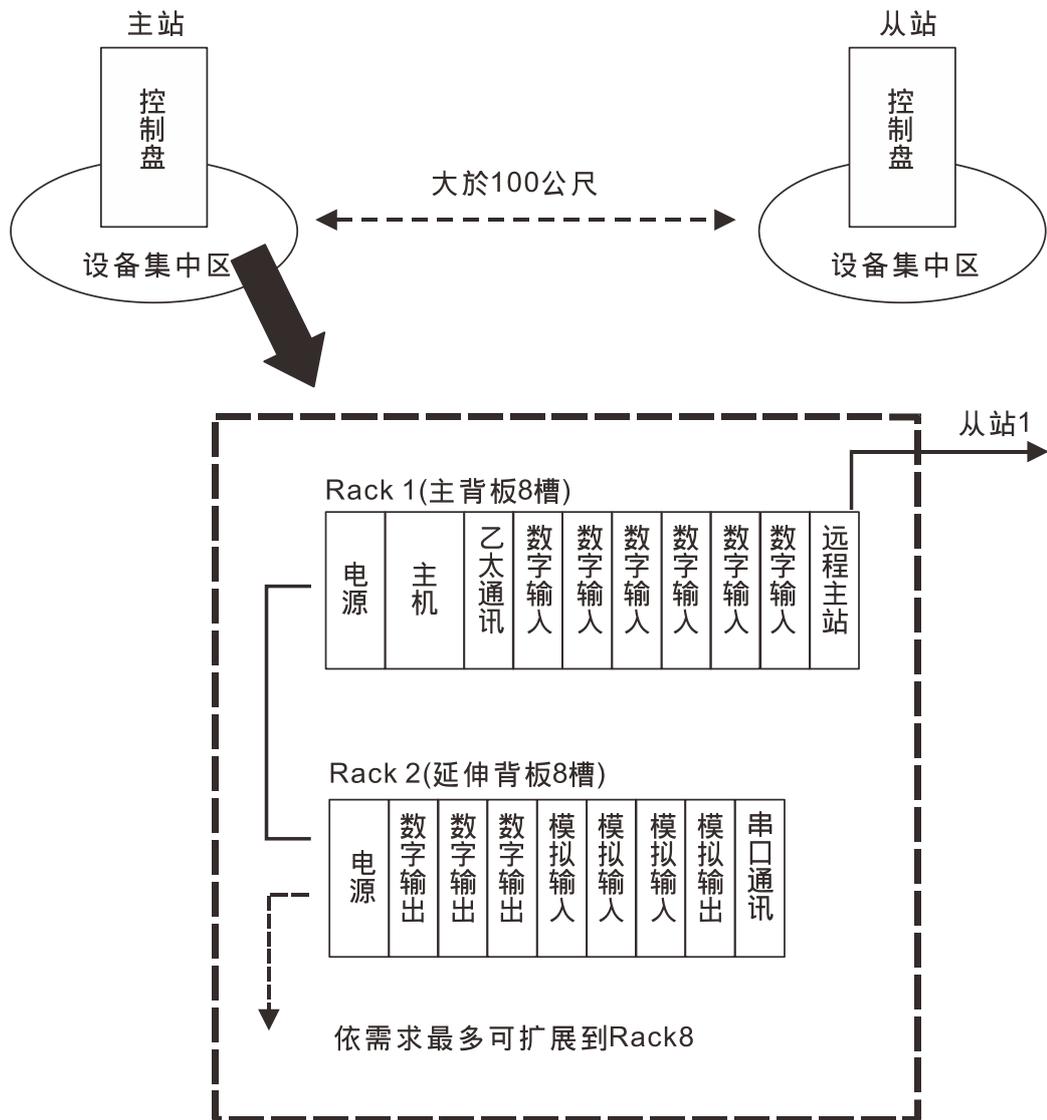
在系统整合 SI 的应用领域里，AH500 可应用于多数的系统控制需求，一般来说，这类的系统架构大都与现场设备的位置分布有关，通常会在设备较为集中的地方放置一个控制盘以节省现场布线成本，但这类集中的地方通常不会只有一个，而且每个集中点之间的距离通常都超过 100 公尺，所以一旦有必要放置两个以上的控制盘时，就会使用到远程 I/O 的系统架构；而 AH500 的远程架构除了具有一般的远程需求之外，也融合了本地 I/O 的扩充便利性，在大部分的情形下，一个控制盘就可以简单视为一个远程 I/O 站，而不是一个背板就视为一个站；除此之外，背板、电源与其他模块也保有共享性，不但增加了系统规划的便利性，也降低了产品选用的困难度。

系统架构：机架 1 (Rack1) ~机架 8 (Rack8) + 远程扩充



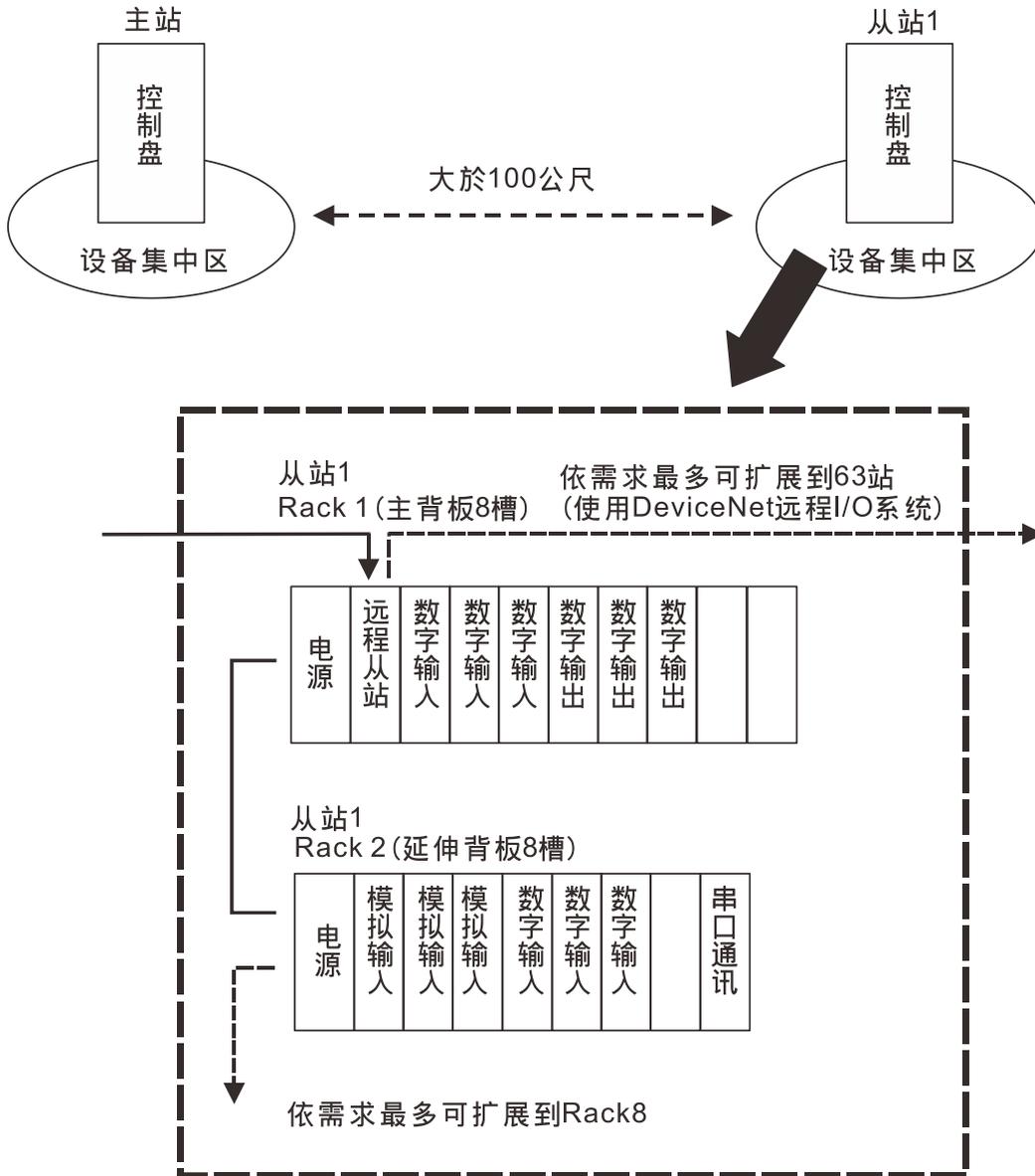
注：扩充背板的延伸，除提供专用电缆(最长可达 100 米)外，另提供光纤电缆(另搭配专用转接器)，长度可达 2km。

规划 1

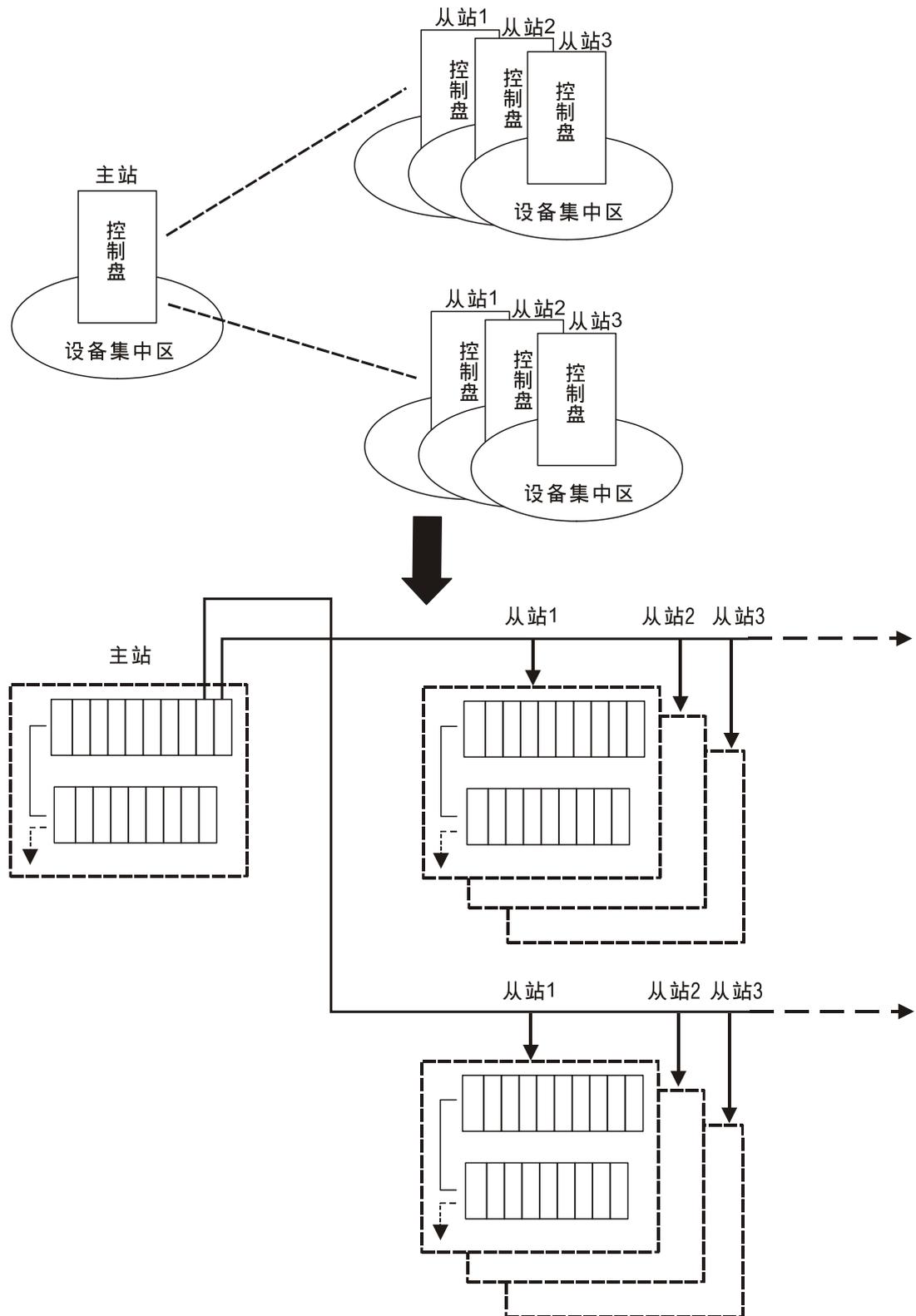


规划 2

1



规划 3

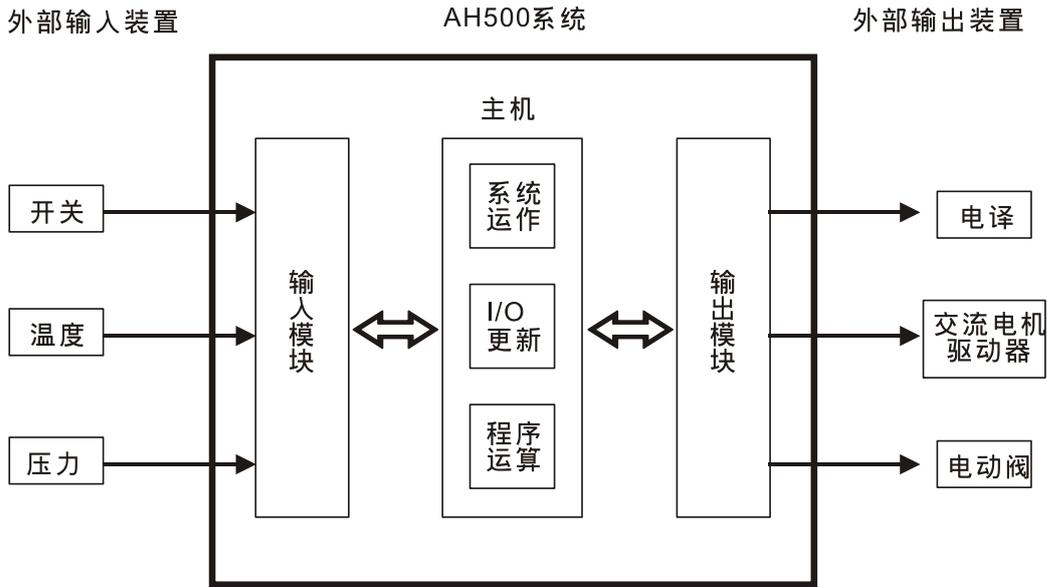


注意：DeviceNet 远程 I/O 的主站模块属于网络模块，所以必须与主机安装在同一个背板，而一个主机可以支持 8 个主站模块（即 8 个通讯主线），每一个主线最多可以规划 63 个从站，每个从站除了主背板之外还可以附加最多 7 个延伸背板。

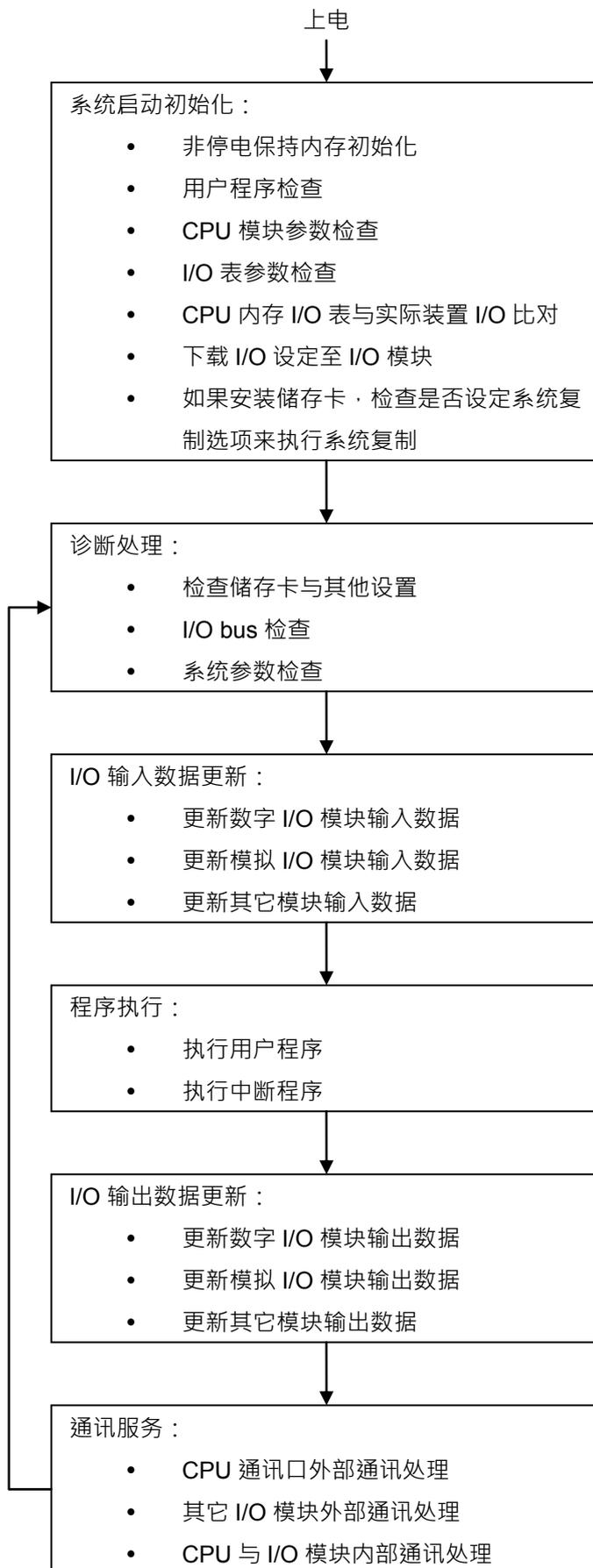
1

1.3 主机运作介绍

主机是整个 AH500 的核心组件，其最主要的工作除了负责执行用户的逻辑程序之外，同时也负责所有 I/O 的数据收送与通讯数据的处理等工作。而主机与相关模块所建立起来的 AH500 系统，与实际外部装置的关系可以简单表达如下：



以上为主机运作简单的表达方式，其中简化了初始化、诊断、通讯等系统面的流程，以及外部中断、时间中断等程序面的流程，若用户有兴趣想更深入的了解，可以参考完整的说明手册内容，以下列出完整的主机运作流程以供参考。



MEMO



2

第2章 程序规划编写

目录

2.1 准备工作	2-2
2.1.1 硬件	2-2
2.1.2 软件	2-3
2.1.3 工具与材料	2-3
2.2 安装	2-4
2.2.1 安装模块	2-4
2.2.2 安装脱落式端子	2-5
2.3 配线	2-7
2.3.1 电源模块配线	2-7
2.3.2 数字输入模块配线	2-9
2.3.3 数字输出模块配线	2-9
2.3.4 模拟输入与输出模块配线	2-10
2.3.5 送电	2-12
2.4 范例说明	2-12
2.5 建立项目	2-14
2.6 规划硬件架构	2-16
2.7 建立全局符号	2-21
2.8 建立功能块	2-23
2.9 建立主要程序	2-31

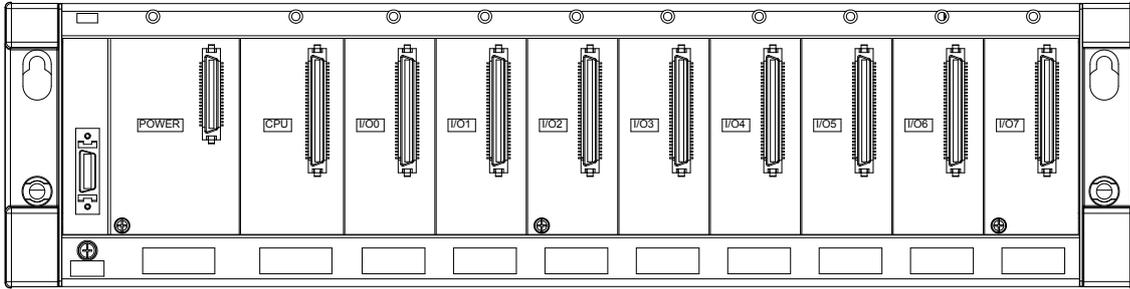
2.1 准备工作

2.1.1 硬件

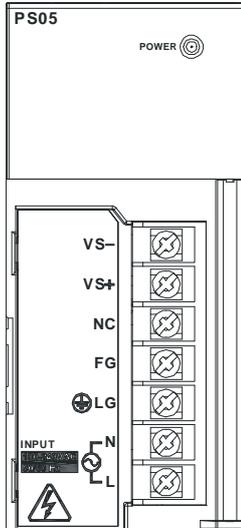
范例所需要的硬件如下：

2

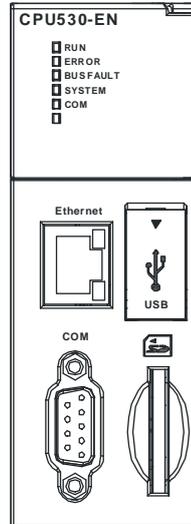
1. AH500 主背板 (8 槽) (AHBP08M1-5A) x 1



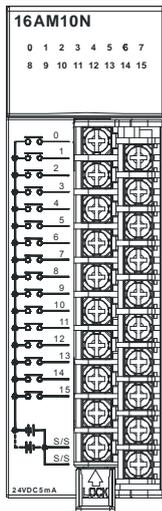
2. AH500 电源模块
AHPS05-5A x 1



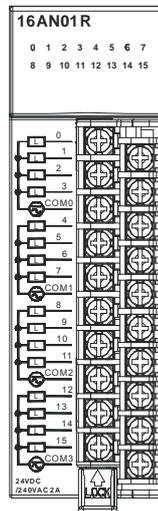
3. AH500 主机 (内建网络功能)
AHCPU530-EN x 1



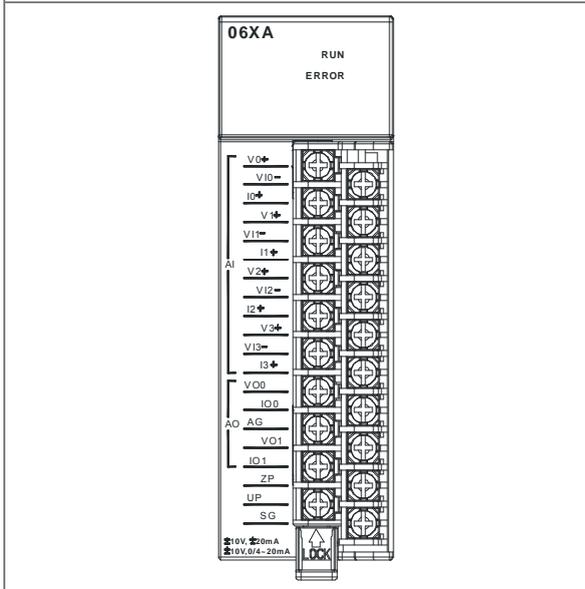
4. AH500 数字输入模块 16 点
AH16AM10N-5A x 1



5. AH500 数字输出模块 16 点
AH16AN01R-5A x 1



6. AH500 模拟输入输出混合模块 (6 通道) AH06XA-5A x 1



2.1.2 软件

范例所需要的软件如下：

- ISPSOft V2.0 以上
- COMMGR V1.0 以上

2.1.3 工具与材料

范例所需要的工具与材料如下：

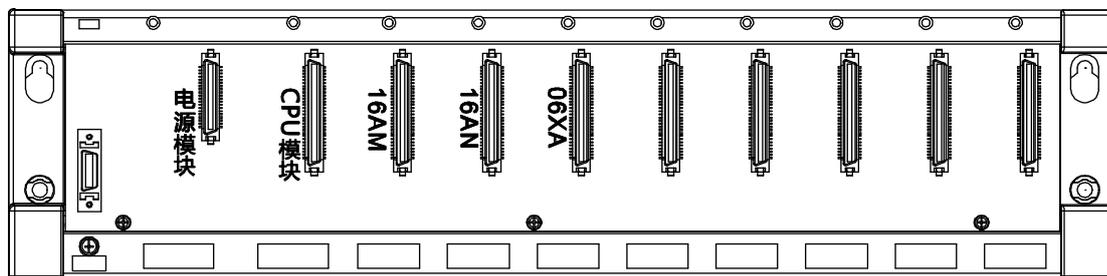
- 个人工作计算机 (已安装上述需求之软件) x 1 台
- 100~240VAC 50/60Hz 的电源供应座 x 1 个
- 24VDC 电源供应器 x 1 个
- 电线 x 1 捆
- 螺丝起子 x 1 支
- USB 连接线或是网络线 x 1 条 (若欲使用 Ethernet 或 COM (RS-232/RS-485) 口接线，请参考 ISPSOft 使用手册第 2.3.2 节；关于 USB 驱动安装说明，请参考 AH500 操作手册附录 A)
- 若有需要可以准备开关与灯泡等配件 (以便模拟外部设备动作) x 1 式

2.2 安装

2.2.1 安装模块

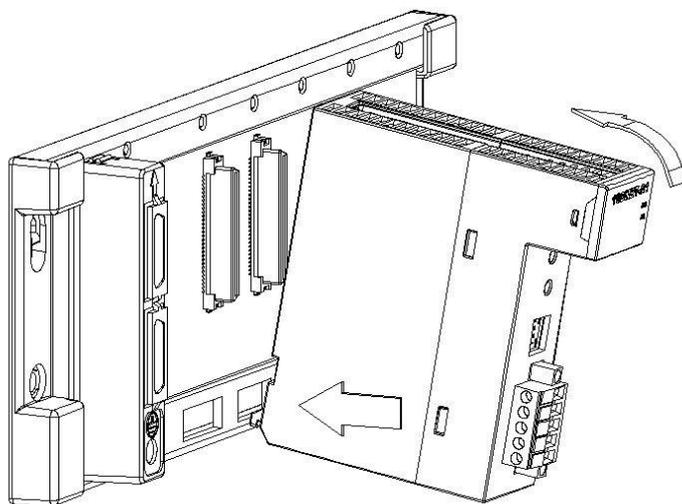
请依照下图的安装配置，将模块安装到主背板对应的槽位上。

2

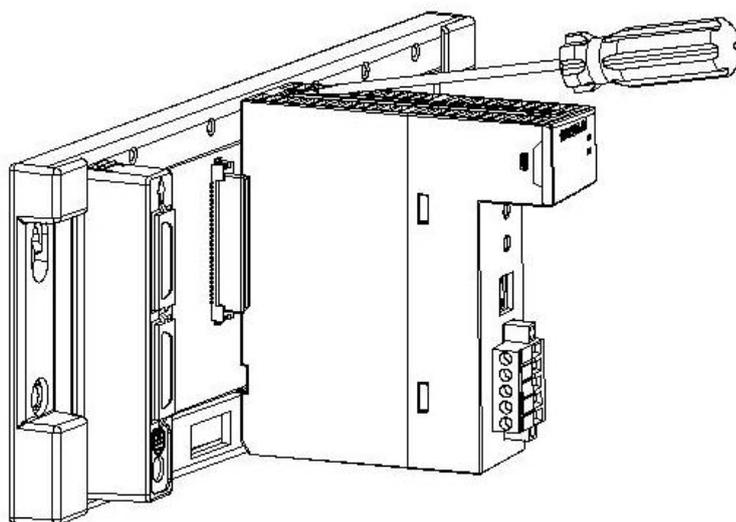


关于模块安装的方式如下图所示，将模块插入插槽中，并且确认模块有妥善地卡住背板，螺丝固定。

1. 将模块下方的卡勾卡在背板的卡槽中。
2. 对准背板的 IO 接口将此模块向前压，如下图所示。



3. 安装到位之后，将此模块上方的螺丝锁紧。

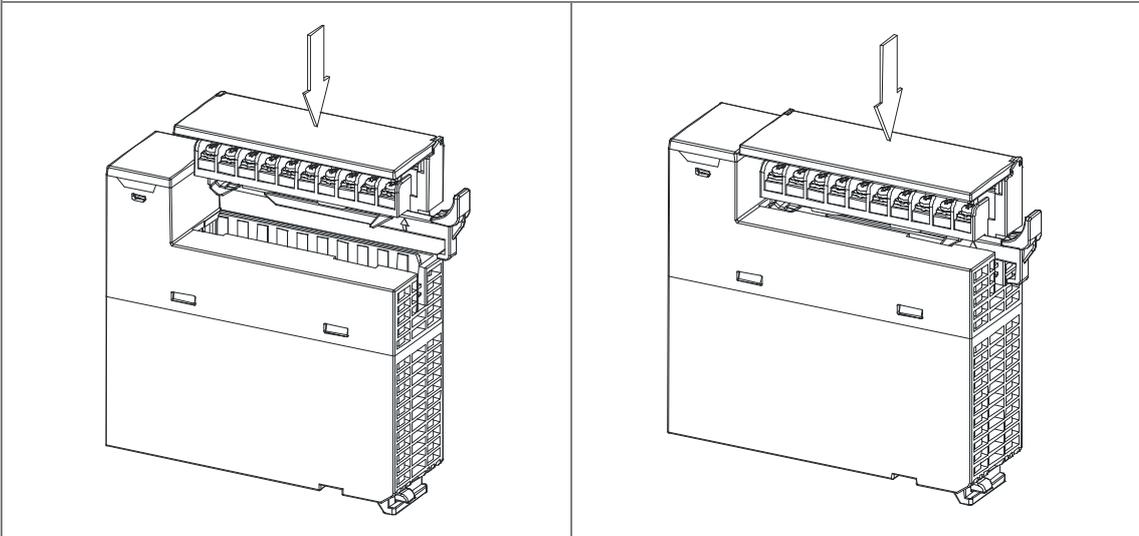


2.2.2 安装脱落式端子

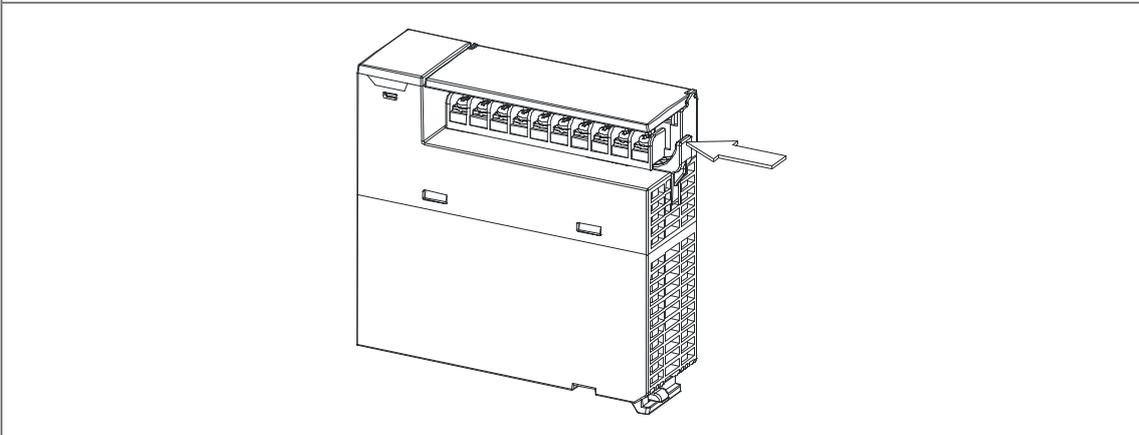
将脱落式端子安装到模块上，安装方式说明如下。

● 脱落式端子安装方法

1. 将端子下方卡槽对准 PCB，往下押入，如下图所示。



2. 将脱落式端子固定杆向内压，即可安装好端子。如下图所示。

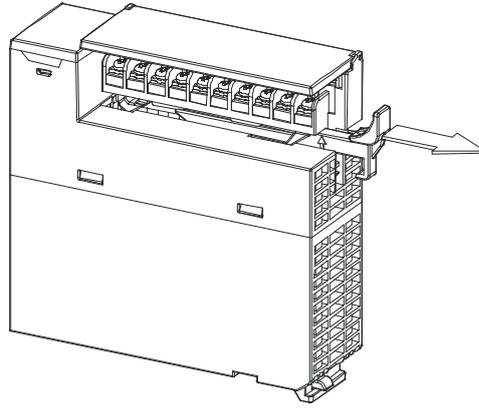
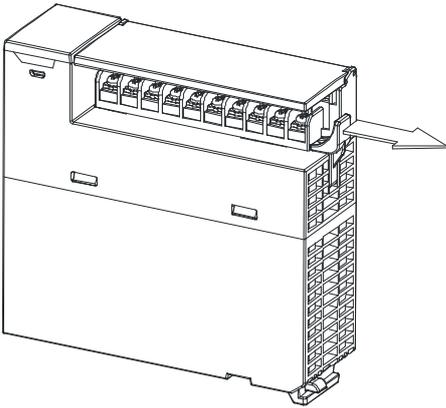


2

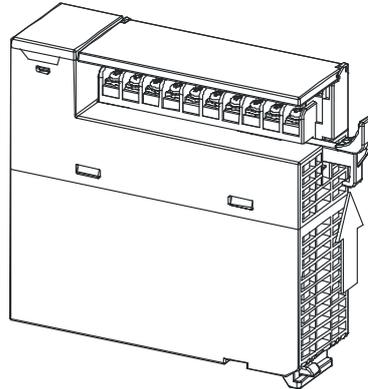
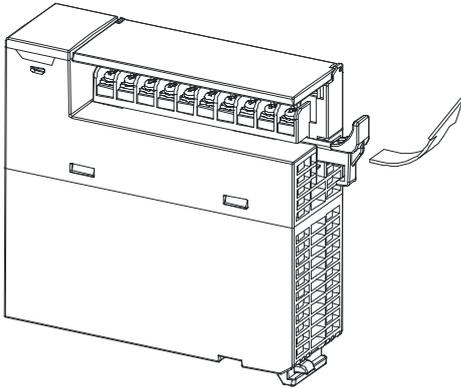
2

● 脱落式端子取出方法

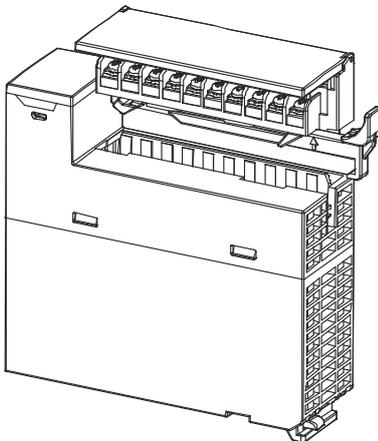
1. 将脱落式端子固定杆向外拉出，如下图所示。



2. 将脱落式端子固定杆向上拉，如下图所示。

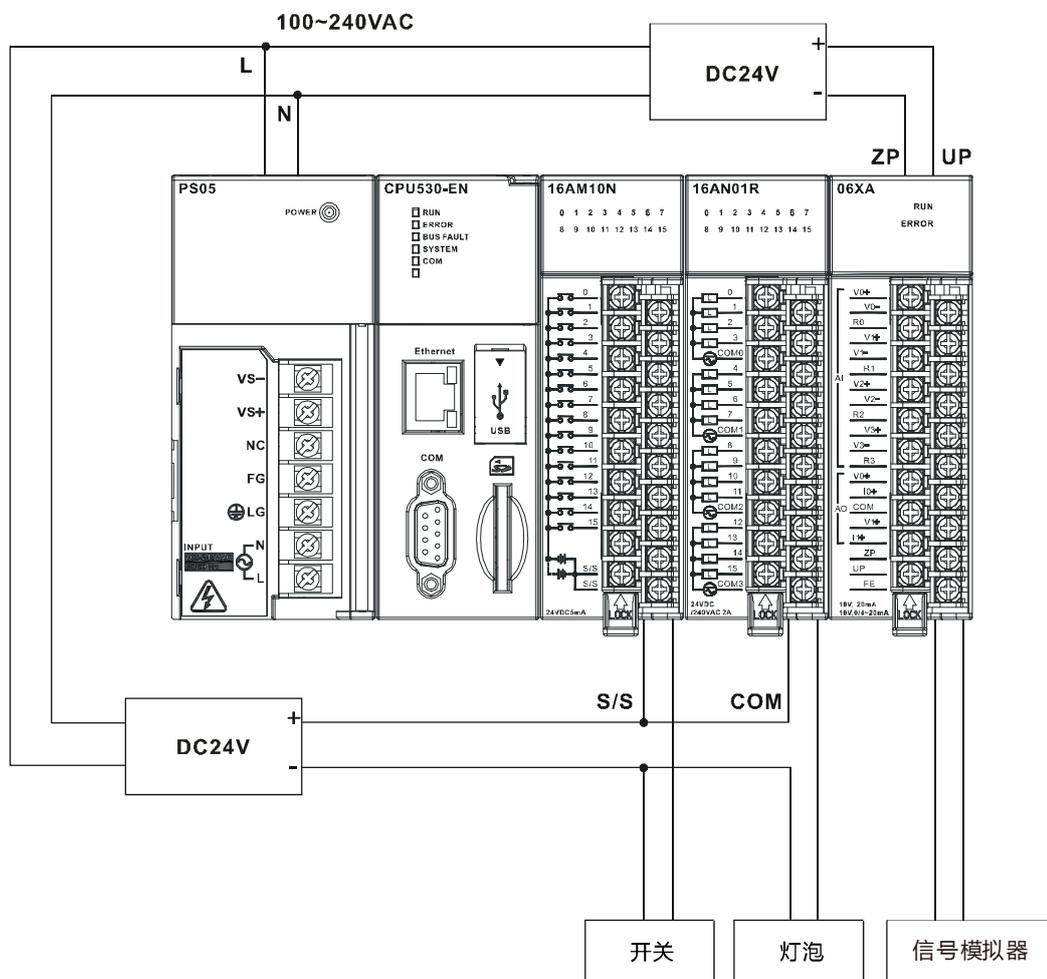


3. 即可取出端子。如下图所示。



2.3 配线

当安装好模块之后，接下来就可以进行模块的接线工作，为了后续范例的顺利进行，至少须将模块电源线的部份接上，也就是电源模块与模拟模块的电源部分，连接电源线的时候，请注意必须在断电的情况下操作以确保安全，而信号线的部份可依个人需求，看是否需要连接到开关与灯泡等组件，以增加操作仿真的便利性与真实感，大致的架构如下：



模块配线说明如下（较完整的说明与注意事项请参阅 AH500 硬件手册）。

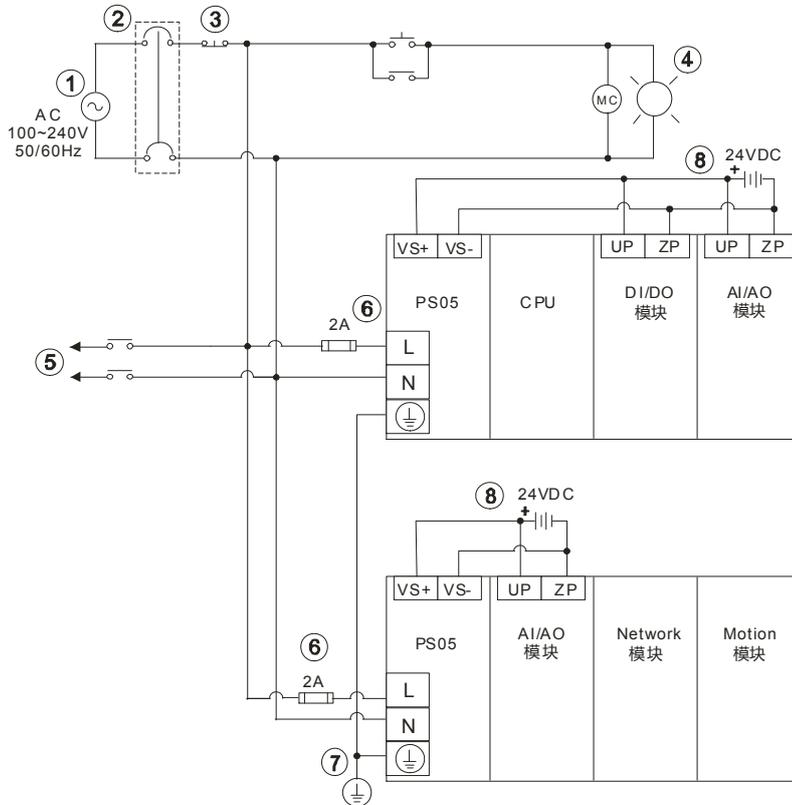
2.3.1 电源模块配线

- 交流电源输入电压，范围宽广（100~240VAC），电源请接于L、N两端，如果将AC110V或AC220V接至VS+与VS-输入端，将使PLC损坏，请用户特别注意。
- 为了确保外部电源稳定提供24VDC，可将外部电源24VDC并接至VS+与VS-，即可检测外部电源是否低于工作电压，如果低于工作电压时，PLC会得知此情况，用户可以依此动作设计保护程序（相关说明请参考AH500操作手册第6.6节）。
- 电源之接地端使用1.6mm以上之电线接地。
- 当停电时间低于10ms时，PLC不受影响继续运转，当停电时间过长或电源电压下降将使PLC停

2

止运转，输出全部OFF，当电源恢复正常时，PLC亦自动回复运转。(PLC内部具有停电保持的辅助继电器及寄存器，用户在作程序设计规划时应特别注意使用。)

- 配线端请使用12-22AWG单蕊线或多蕊线(线材请使用60/75°C铜导线)，而PLC端子螺丝扭力为5~8kg-cm (4.3~6.9lb-in)。
- 安全配线回路：由于PLC控制许多装置，任一装置的动作可能都会影响其他装置的动作，因此任一装置的故障都可能会造成整个自动控制系统失控，甚至造成危险。所以在电源端输入回路，建议的保护回路配置图如下：

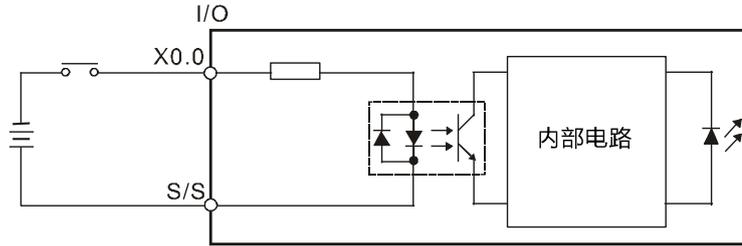


①	交流电源供应：100~240VAC · 50/60Hz
②	断路器
③	紧急停止：为预防突发状况发生，设置紧急停止按钮，可在状况发生时，切断系统电源。
④	电源指示灯
⑤	交流电源负载
⑥	电源回路保护用保险丝 (2A)
⑦	接地阻抗 100Ω 以下
⑧	直流电源供应：24VDC

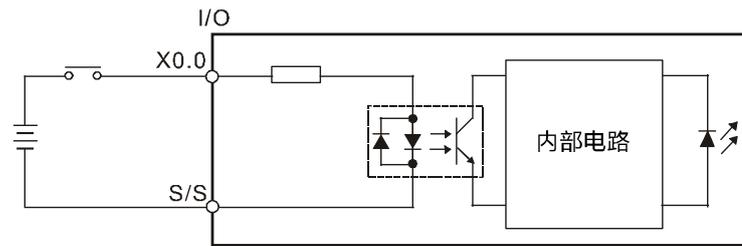
2.3.2 数字输入模块配线

输入点之入力信号为直流电源DC输入，DC型式共有两种接法：漏型模式(Sink)及源型模式(Source)，其定义如下：

- 漏型模式 (Sink)



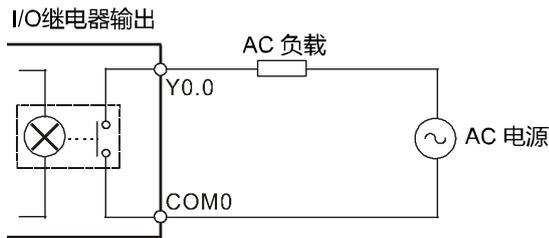
- 源型模式 (Source)



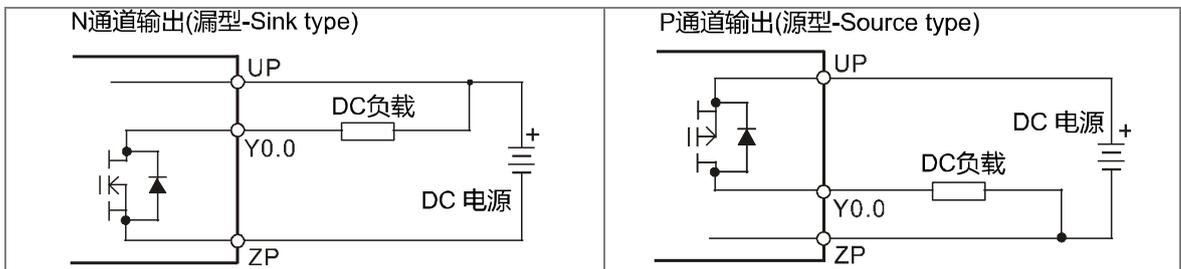
2.3.3 数字输出模块配线

输出组件有继电器、晶体管及TRIAC三种

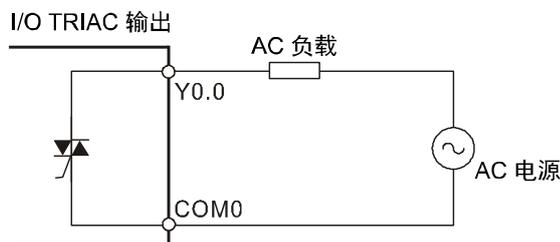
继电器输出 (本手册选用的模块为此类型)



晶体管输出



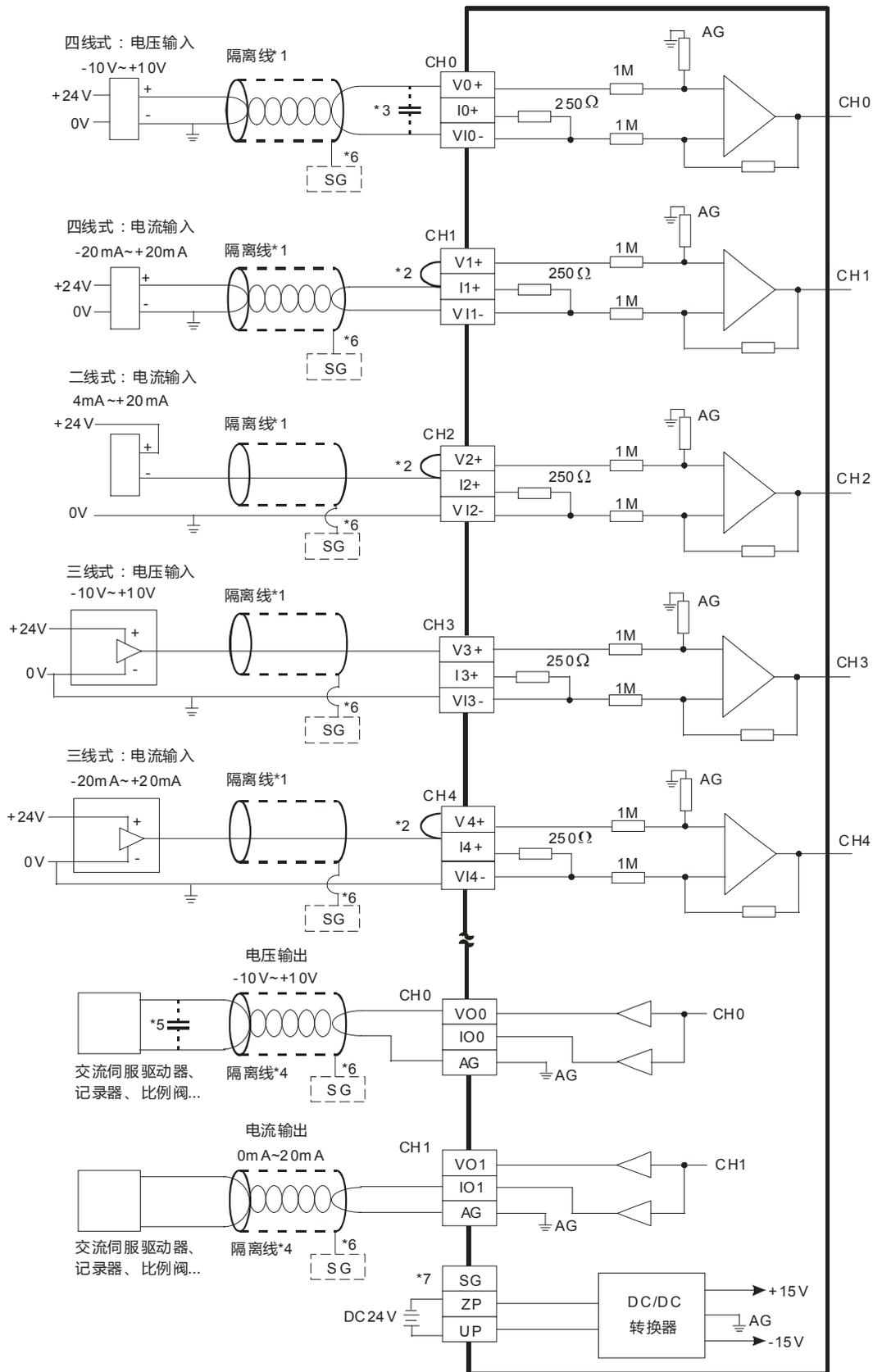
TRIAC输出



2

2.3.4 模拟输入与输出模块配线

2



- *1. 模拟输入信号线请与其它电源线隔离。
- *2. 如果连接电流信号时，V+及I+端子请务必短路。
- *3. 如果输入电压有涟波造成配线受噪声干扰时请连接0.1~0.47 μ F 25V之电容。
- *4. 模拟输出信号线请与其它电源线隔离。
- *5. 如果负载之输入端涟波太大造成配线受噪声干扰时，请连接0.1~0.47 μ F 25V之电容。
- *6. 请将隔离线接地端接至SG端子。
- *7. 当模块正确地安装在背板上，则SG已直接与背板之 ⏚ 端短接，并将背板之 ⏚ 端连接至大地 ⏚ 端。



2.3.5 送电

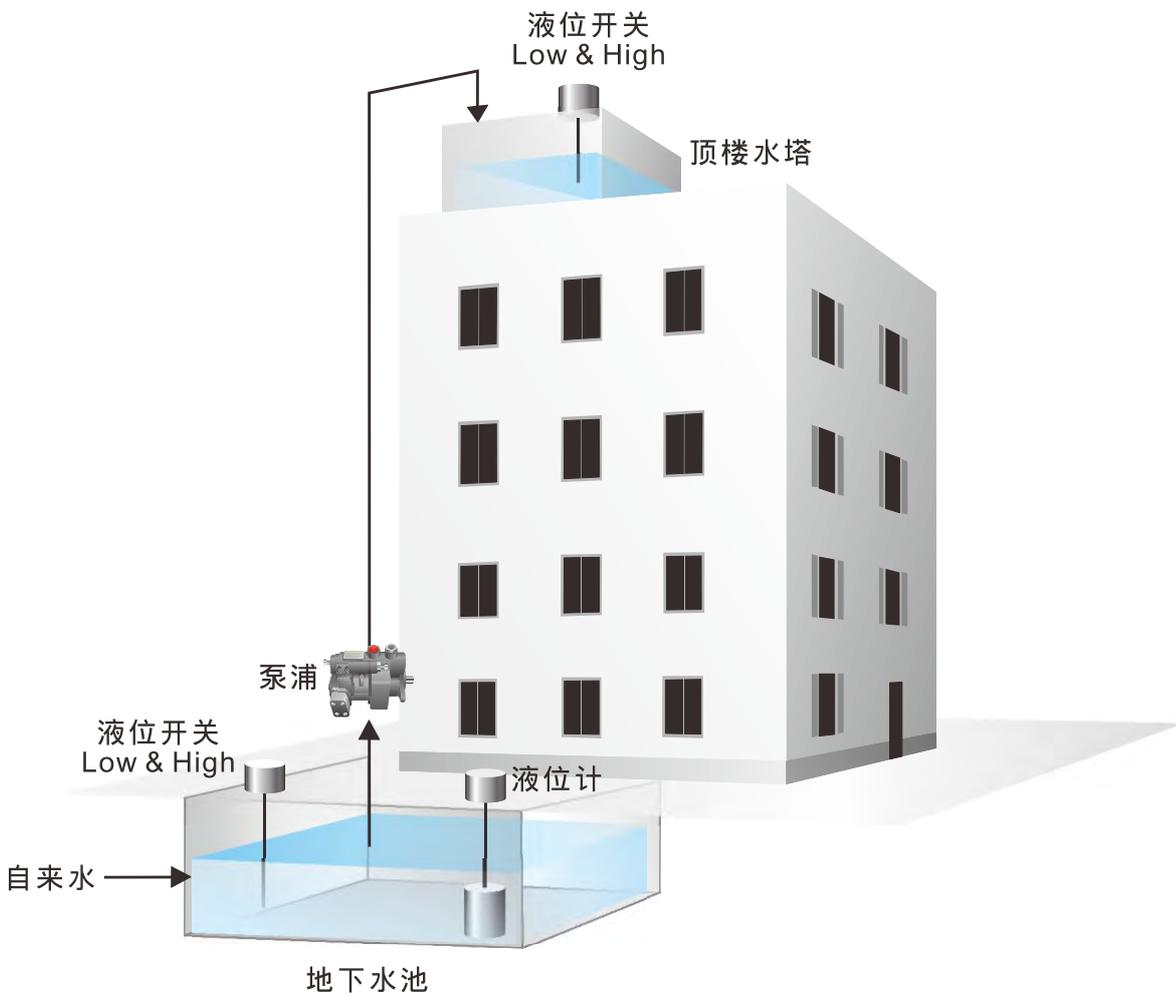
当所有配线确认无误完成之后，就可以准备送电，送电之前请先将主机的开关设置在STOP的位置，同时送电之后，主机会开始进行初始化的检查动作，当检查动作完成后，因为全新出厂的主机内部是没有任何硬件规划的，所以会出现错误灯号，这是属于正常的现象可以暂时先忽略。

2

2.4 范例说明

当硬件安装好并完成配线与送电之后，接下来就可以开始准备进行程序编写的工作，在真正开始编写之前，为了让用户有一个明确的思考目标与方向，本手册准备了一个常用实例，并藉由一步一步的操作方式，将一个项目的完整流程，从建立新项目到下载至主机执行完整演练一便，而以下是本范例的内容与架构说明。

- 系统架构示意



● 控制动作需求

本范例为一般大楼标准供水系统的基础设计，自来水会自动补入地下水池，而地下水池的水可经由泵浦输送至顶楼的水塔内，然后利用重力分送到各楼层使用，而泵浦的起停控制是依据地下水池与顶楼水塔的液位开关来动作的，同时为了监测自来水的供水情形，在地下水池部分另加装了压力式液位计，以便随时监视地下水池的蓄水容量。

PLC I/O 连接组件与装置：

1. 单点式液位开关 (A 接点) x 1
装置于地下水池，信号 (Low) 接点部份会连接至数字输入模块。
2. 两点式液位开关 (A 接点) x 1
装置于顶楼水塔，信号 (Low & High) 接点部份会连接至数字输入模块。
3. 泵浦 x 1
装置于地下水池附近，但实际 PLC 连接的装置并非为泵浦本体而是控制盘，而以一般的通用设计，通常为 3DI/1DO 的点数，远程控制 (Remote) x 1 (DI) & 运转 (Run) x 1 (DI) & 异常 (Trip) x 1 (DI) & 启动 (Start) x 1 (DO)，而这些信号会连接至数字输入与数字输出模块。
4. 压力式液位计 x 1
装置于地下水池，水位信号 (Level) 接点部份会连接至模拟输入模块 (0~10V 对应水位深度 0.0~10.0 公尺，0V 代表 0.0 公尺、10V 代表 10.0 公尺)。

泵浦控制条件：

1. 启动：当顶楼水塔的液位 Low 触发时 (代表顶楼水不够)，且地下水池的液位 Low 并未触发时 (代表地下水池水足够，可以提供)，则启动泵浦进行补水。
2. 停止：当顶楼水塔的液位 High 触发时，或是地下水池的液位 Low 触发时，则停止泵浦运转。

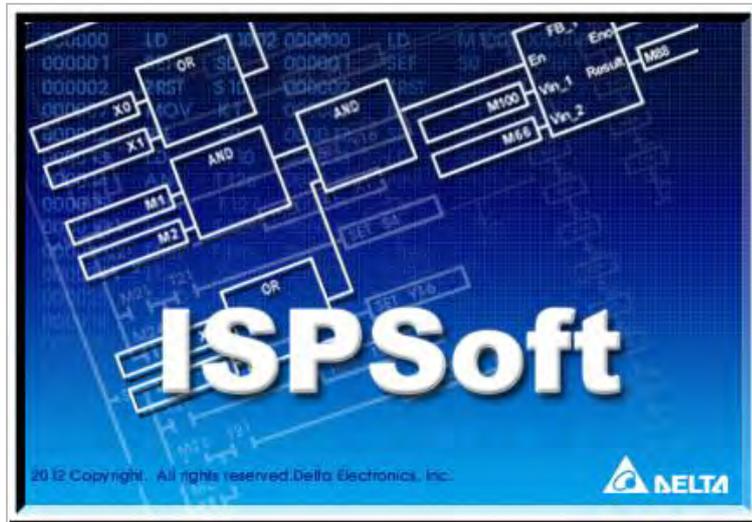
2.5 建立项目

当硬件安装完成并了解范例内的控制逻辑后，即可开始准备编写程序。

步骤 1：开始当然是要开启 PC 端 PLC 编辑软件 ISPSOft (开始 → 程序集 → Delta Industrial Automation → PLC → ISPSOft.xx → ISPSOft.xx)。

2

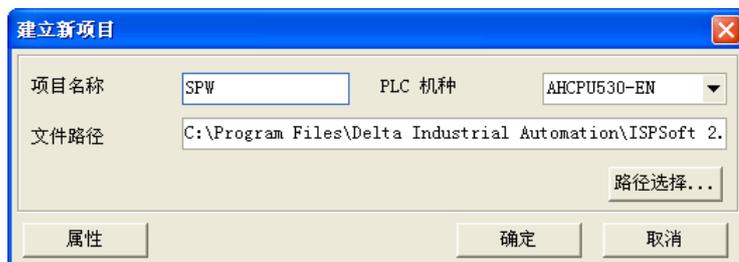
开启画面



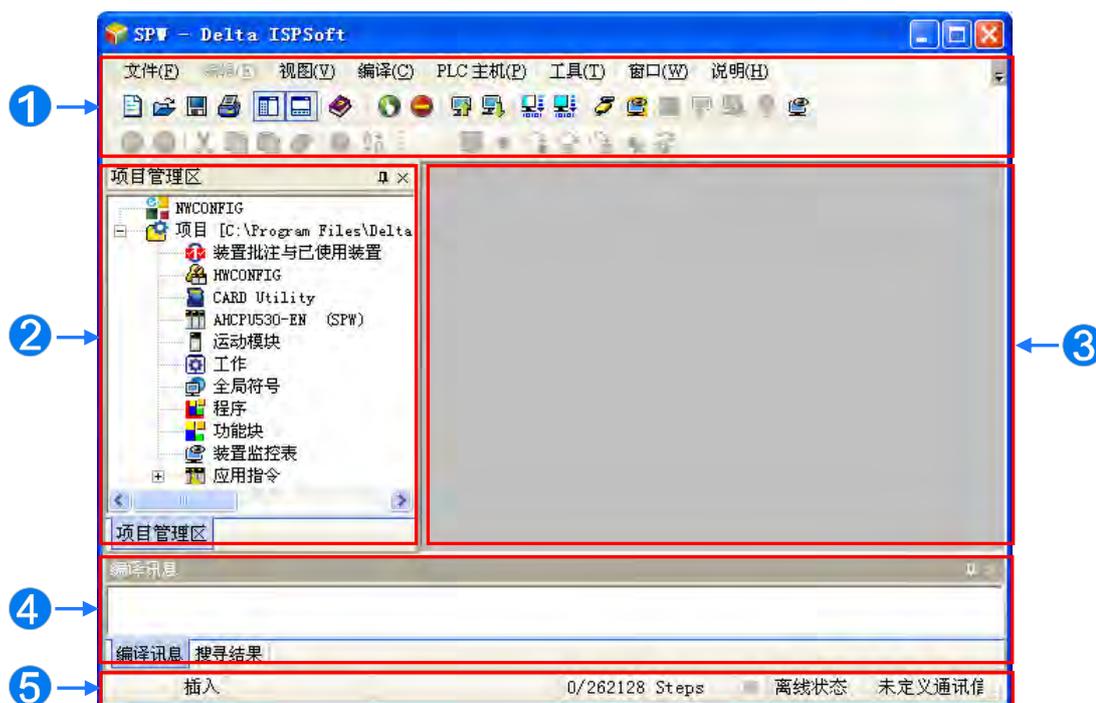
进入软件主画面



步骤 2：接着按下新项目的按键 () 就可以新增一个项目。而这个范例与供水系统有关，所以我们这个范例的名称就使用 SPW 的项目名称，PLC 机种就选择之前我们选定的 AHCPU530-EN 的型式，至于文件路径部份就暂时先以默认路径储存，最后按下确定键。



接下来就会显示整个项目主环境 (左上角会显示项目名称 SPW)，如此一个项目就算建立完成。



软件操作接口简易说明

- ① 功能区：包含了整个编辑软件环境的主要功能，很多常用的功能都会放置在此区的快捷功能列上，而比较细项的功能则会整理在功能列里面。
- ② 项目管理区：这里展示了整个项目的主要对象架构，由这样的树状架构可以很容易的理解项目对象的关系，也加强了项目功能管理的效率。
- ③ 工作区：大部分用户的编辑工作都会在此区。
- ④ 输出区：部份功能执行时会产生一些信息，而这些信息就会显示在这个区域，以帮助用户编辑。
- ⑤ 状态区：显示目前软件的一些项目与通讯信息。

2.6 规划硬件架构

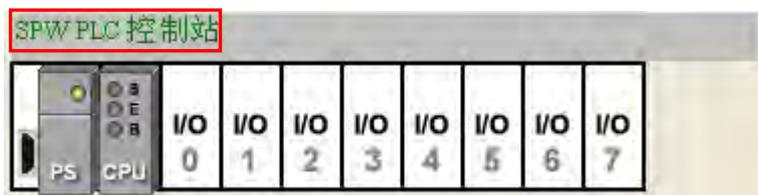
当项目建立完成后，就可以进行硬件的规划工作，针对这个范例我们先假设以下的规划。

- 16AM 数字输入的部份 (16 点) 规划地址为 X0.0 ~ X0.15
- 16AN 数字输出的部份 (16 点) 规划地址为 Y0.0 ~ Y0.15
- 06XA 模拟输入的部份 (4 通道) 规划地址为 D0 ~ D7
- 06XA 模拟输出的部份 (2 通道) 规划地址为 D100 ~ D103
- 地下水池的 Low 点 → X0.0
- 顶楼水塔的 Low 点 → X0.2
- 顶楼水塔的 High 点 → X0.3
- 泵浦的远程控制 (Remote) 点 → X0.5
- 泵浦的运转 (Run) 点 → X0.6
- 泵浦的异常 (Trip) 点 → X0.7
- 泵浦的启动 (Start) 点 → Y0.0
- 地下水池的 Level 值 → D0

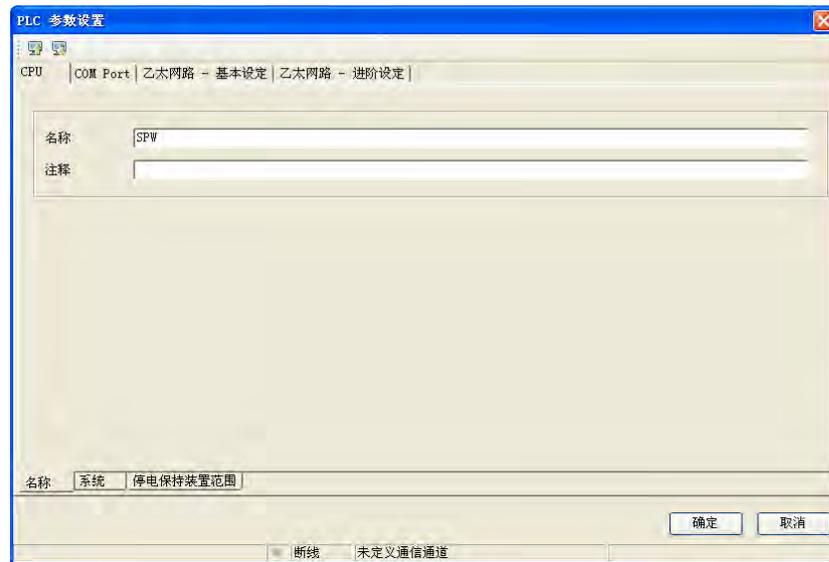
有了以上的信息后就可以实际进行规划了，首先可以在软件主画面左侧的窗口里找到 HWCONFIG 的图标与文字  HWCONFIG，然后直接连点两下鼠标左键就可以开启规划窗口。



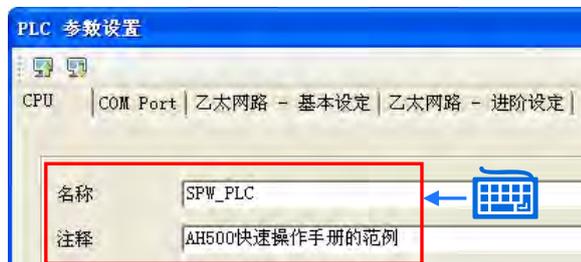
现在就可以依照之前所规划的范例信息开始进行规划了，首先建议输入批注以便项目分享，在这我们输入以下文字内容「SPW PLC 控制站」。



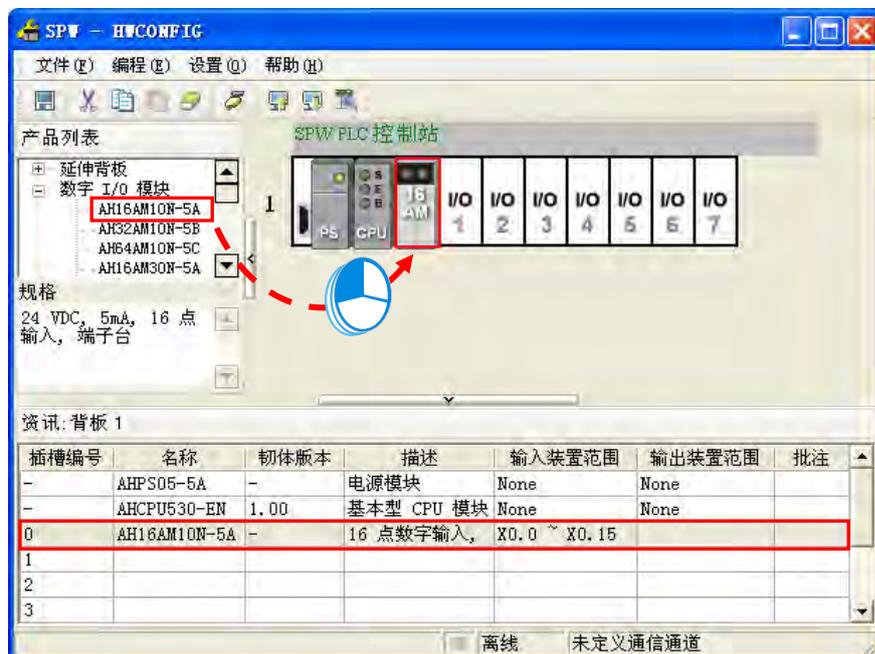
再来连点两下 CPU 模块，开启主机设定窗口。



请在名称的部份填入「SPW_PLC」当做主机识别使用，批注的部份可以依需求填入「AH500 快速操作手册的范例」，其余部份可以保持默认值不需要变更，然后按下「确定」，关闭此交谈窗口。

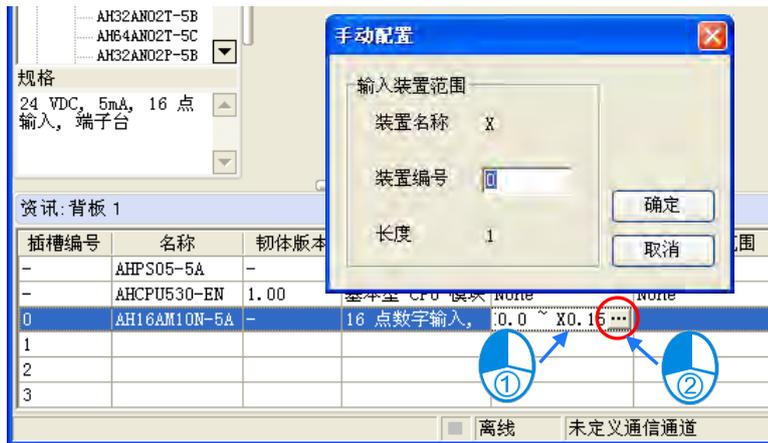


接下来可以进行模块放置的动作，先到左侧的产品列表窗口，找到我们需要的第一个模块 AH16AM10N-5A，并利用鼠标拖曳的方式，将模块拖曳到 I/O 0 的位置后放开。



模块放置后，系统会自动配置地址，默认值都是由 0 开始，所以默认地址为 X0.0~X0.15，刚好符合我们的需求不需要修改，若不符合需求也可以利用鼠标点击该字段以进行修改。

2



依照同样的方式将 AH16AN01R-5A 置入 I/O 1 的位置，地址规划为 Y0.0~Y0.15。



依同样方式将 AH06XA-5A 置入 I/O 2 的位置，地址规划为 IN : D0~D7 / OUT : D100~D103。
 (输出装置范围的默认值为连续地址 D8~D11，请记得修改)

2



除了指定装置范围之外，模拟量 IO 模块尚须指定对应的固件版本，请依实际机器的版本选择即可。

插槽编号	名称	固件版本	描述	输入装置范围	输出装置范围	批注
-	AHPS05-5A	-	电源模块	None	None	
-	AHCPU530-EN	1.00	基本型 CPU 模块	None	None	
0	AH16AM10N-5A	-	16 点数字输入,	X0.0 ~ X0.15		
1	AH16AN01R-5A	-	16 点数字输出,		Y0.0 ~ Y0.15	
2	AH06XA-5A	1.00	4 通道模拟输入	D0 ~ D7	D100 ~ D103	
3		0.38				
4		0.41				
5		1.00				

在模拟模块的部份，必须还要设定信号类型与工程值转换关系，这样才算是完成基本设定，如果要进行这样的设定，可以直接连点该模块图标两下，系统会自动开启对应的参数设定窗口。



依照假设需求，调整信号为 0~10V 电压工作模式。

2



依照假设需求，调整工程值转换关系为 0.0~10.0 (M)，其他设定可保持预设然后按下确定。



这样硬件架构部份规划完成，请存盘后关闭窗口离开，回到软件项目主画面。



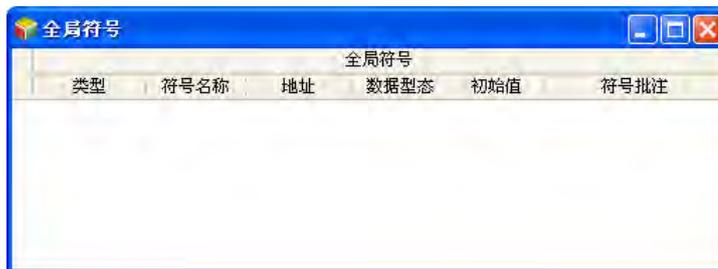
2.7 建立全局符号

为了程序的可读性以及未来连接 SCADA 系统的便利性，一般会将实体 I/O 的地址部份再配置一个全局符号，这样在编写程序时，就可以使用容易理解的符号方式来编辑，同时全局符号表也支持汇入导出的功能，针对中大型系统点数较多的情形，操作时可以利用 Excel 等软件来让编辑更便利与更快速。

在这个例子我们会建立以下的全局符号：

全局符号总表		
硬件点 (for PLC I/O)		
地址	符号	数据类型
X0.0	Tank_B1F_LSW	BOOL
X0.2	Tank_RF_LSW	BOOL
X0.3	Tank_RF_HSW	BOOL
X0.5	SPP01_Remote	BOOL
X0.6	SPP01_Run	BOOL
X0.7	SPP01_Trip	BOOL
Y0.0	SPP01_Start	BOOL
D0	Tank_B1F_LT	REAL
软件点 (for SCADA)		
M0	SPP01_Auto	BOOL
M1	SPP01_Man_SW	BOOL

有了以上的信息后，我们可以准备开始建立全局符号，首先在主画面左侧窗口「项目管理区」可以找到「全局符号」的图标与文字  **全局符号**，连点两下该图标或文字就会开启以下编辑窗口。



接着在空白处连点两下鼠标，系统会跳出信息编辑接口，接着请依照表格内容先建立第一点「X0.0」，其中关于类别、初始值、符号批注等字段可以维持默认值即可，然后按下确定的按键完成输入。



这样就可以看到有一笔新数据产生。

全局符号					
类型	符号名称	地址	数据类型	初始值	符号批注
VAR	Tank_B1F_ISW	X0.0	BOOL	N/A	地下水池的液位开关 Low Switch

2

依照同样的方式，将表格内所有的数据建立在全局符号内，完成后应如下图所示。

(模拟的对应部分是使用 REAL 浮点数格式，所以一个符号是占用 2 个 D 装置，而字段内所填的地址为起始地址。)

全局符号					
类型	符号名称	地址	数据类型	初始值	符号批注
VAR	Tank_B1F_LSW	X0.0	BOOL	N/A	地下水池的液位开关 Low Switch
VAR	Tank_RF_LSW	X0.2	BOOL	N/A	顶楼水塔的液位开关 Low Switch
VAR	Tank_RF_HSW	X0.3	BOOL	N/A	顶楼水塔的液位开关 High Switch
VAR	SPP01_Remote	X0.5	BOOL	N/A	地下供水 泵浦 SPP01 Remote
VAR	SPP01_Run	X0.6	BOOL	N/A	地下供水 泵浦 SPP01 Run
VAR	SPP01_Trip	X0.7	BOOL	N/A	地下供水 泵浦 SPP01 Trip
VAR	SPP01_Start	Y0.0	BOOL	N/A	地下供水 泵浦 SPP01 Start
VAR	Tank_B1F_LT	D0	REAL	N/A	地下水池液位 LT
VAR	SPP01_Auto	M0	BOOL	N/A	地下供水 泵浦 SPP01 Auto Mode
VAR	SPP01_Man_SW	M1	BOOL	N/A	地下供水 泵浦 SPP01 Manual输出

2.8 建立功能块

本范例的流程是先建立功能块的程序部份，当然也有人喜好先建立主流程的程序部份，这并没有绝对的关系，而使用的经验上，其实两者常常交互使用开发并无绝对的顺序，不过建议应当先开发功能块的部份会比较便利，尤其是常常重复使用的功能，就建议使用功能块节省程序开发时间。

以本范例来看，就会考虑把水池、水塔、与泵浦的控制关系写成功能块 (FB)，因为实际的应用上，一个大楼并不会只有一套这样的供水系统，通常会有两套以上的设计，如果使用功能块的方式，则在程序上只需要更换功能块 Input 与 Output 接脚的变量，就等于完成第二套系统的开发。

现在我们就开始建立功能块，目的就是把水池、水塔、与泵浦三者的控制关系包在一个功能块里面，首先在主画面左侧的窗口「项目管理区」内找到「功能块」的图标与文字 ，利用鼠标右键的方式选择新增一个 POU。



填入 POU 名称与注释，内容请参考下图，其余可保持默认值并按下确定。



这样在主画面就可以看到功能块的程序编辑窗口。



首先，先建立功能块的「区域符号」，这个部份就等于将来这个功能块的参数引入与引出（接脚）的部份，建立方式与之前提到的全局符号方式相同，唯一不同的是在地址部份，系统会自动配置，所以不允许用户自行输入，这部份的考虑是为了功能块的可移植性而设计，当然在程序内部仍然可以使用地址或全局符号来编辑程序，但是请注意这样使用时会降低这个功能块的可移植性与便利性。

（区域符号与其他区域符号或是全局符号之间，其符号名称是可以重复的，当重复时系统会以程序或功能块本身的区域符号为识别优先）

以下为本范例需要建立的区域符号：

类别	符号	数据类型
VAR_INPUT	Tank_B_LSW	BOOL
VAR_INPUT	Tank_R_LSW	BOOL
VAR_INPUT	Tank_R_HSW	BOOL
VAR_INPUT	Pump_Remote	BOOL
VAR_INPUT	Pump_Run	BOOL
VAR_INPUT	Pump_Trip	BOOL
VAR_IN_OUT	Pump_Auto	BOOL
VAR_IN_OUT	Pump_Man_SW	BOOL
VAR_OUTPUT	Pump_Start	BOOL
VAR	Pump_Out	BOOL

VAR_INPUT

程序执行时会先将外部变量的数值引入到内部变量里，当对应的内部变量数值有变更时，并不会再传送给外部变量，通常会使用在外部变量不应该被修改的情形，本范例中大多是由数字输入（DI）而来的状态，像这一类外部变量的状态是不应该被修改的，所以宣告成 VAR_INPUT 可避免程序误改到这类变量的数值，而影响到后续其他程序或功能块的使用。

VAR_IN_OUT

程序执行时会先将外部变量的数值引入到内部变量里，当程序结束时会将再传送给外部变量，通常会使用在变量需要被修改的情形，像本范例中 Pump_Auto 与 Pump_Man_SW 的两个符号变量，一般来说是给 SCADA 系统的用户操作使用，以决定泵浦的控制模式是要自动模式（依液位信息控制）或是手动模式（依照 Pump_Man_SW 的开关状态控制），看起来好像只要 VAR_INPUT 类别就可以，但是考虑到当泵浦发生故障（TRIP）时，为了保护与提醒作用，需要将泵浦的模式自动切换到手动且停止命令输出的状态，这样就必须要变更 Pump_Auto 与 Pump_Man_SW 这两个符号变量，所以需要将这两个变量宣告成 VAR_IN_OUT 的类别。

VAR_OUTPUT

程序执行时并不会先将外部变量的数值引入到内部变量里，而是直接使用之前内部记忆的数值，而在程序结束时会将内容再传送给外部变量，一般来说此类变量在程序内使用时，都是落在指令的 OUT 位置，像本范例中的 Pump_Start 就属于这类的使用。

VAR

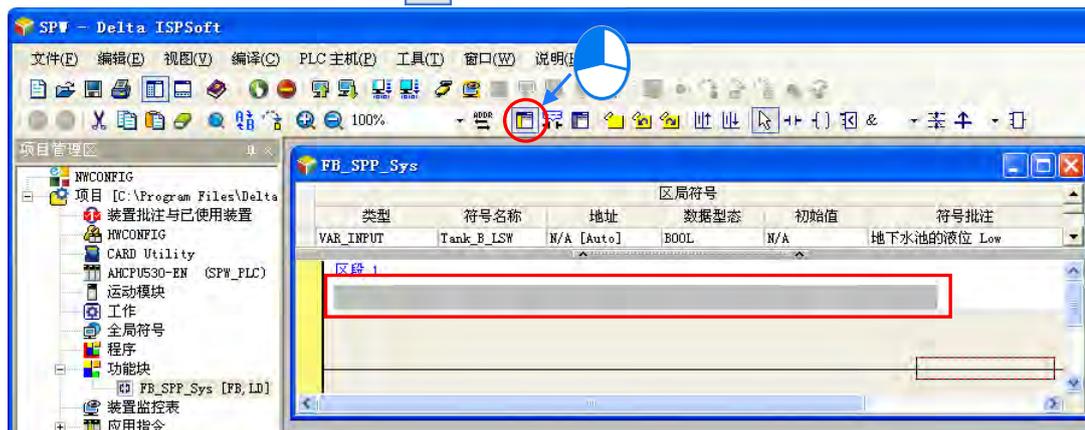
程序执行时会视为内部变量，与 VAR_OUTPUT 一样会使用之前内部记忆的数值，没有引入与传送外部变量的功能，一般来说此类变量在程序内使用时，都是当作程序运算暂存使用，像本范例中的 Pump_Out 就属于这类的使用。

（在程序（Program）内使用功能块（FB）时，若配置同样的符号变量（FB 数据类型）给多次调用使用的功能块时，则不保证此两类型（VAR_OUTPUT 与 VAR）变量的初始数值会是前一次离开功能块时的数值）

当建立完成后应如下图所示。

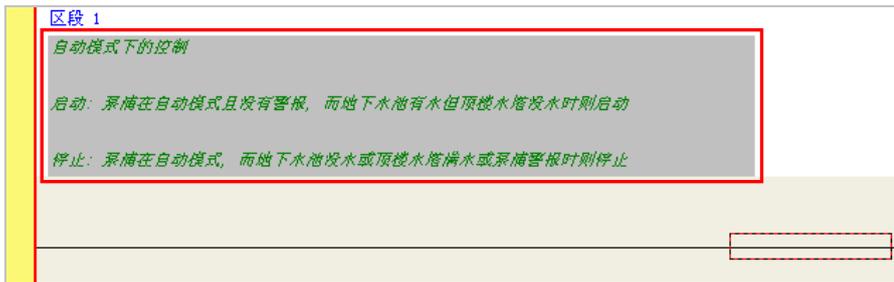
区局符号						
类型	符号名称	地址	数据类型	初始值	符号批注	
VAR_INPUT	Tank_B_LSW	N/A [Auto]	BOOL	N/A	地下水池的液位 Low	
VAR_INPUT	Tank_R_LSW	N/A [Auto]	BOOL	N/A	顶楼水塔的液位 Low	
VAR_INPUT	Tank_R_HSW	N/A [Auto]	BOOL	N/A	顶楼水塔的液位 High	
VAR_INPUT	Pump_Remote	N/A [Auto]	BOOL	N/A	泵浦 Remote	
VAR_INPUT	Pump_Run	N/A [Auto]	BOOL	N/A	泵浦 Run	
VAR_INPUT	Pump_Trip	N/A [Auto]	BOOL	N/A	泵浦 Trip	
VAR_IN_OUT	Pump_Auto	N/A [Auto]	BOOL	N/A	泵浦 Auto	
VAR_IN_OUT	Pump_Man_SW	N/A [Auto]	BOOL	N/A	泵浦 Manual Switch	
VAR_OUTPUT	Pump_Start	N/A [Auto]	BOOL	N/A	泵浦 Start	
VAR	Pump_Out	N/A [Auto]	BOOL	N/A	泵浦 Out	

接着可以准备编写功能块的程序内容了，为了程序的浏览方便建议培养批注的编写习惯，若要编写程序区段批注请先开启区段批注的功能  显示区段批注。

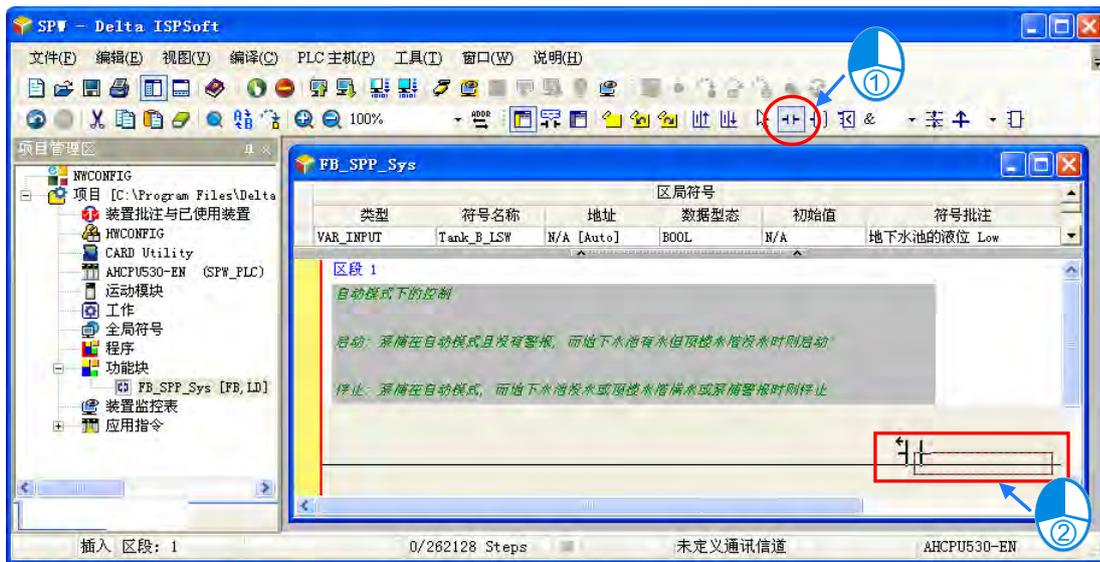


先在区段 1 的批注处填写说明。输入时，按下键盘的【Shift】+【Enter】即可换行。

2

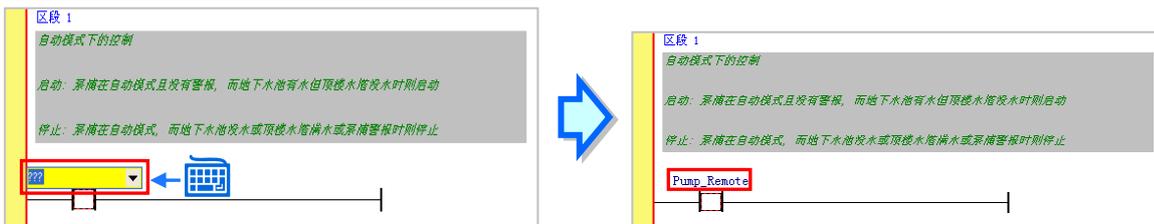


接着编辑程序代码，此处使用的语言为 LAD，请按下上方的 图标后将鼠标移到红框位置，移动鼠标调整放置位置，使光标出现 接点串接符号后，按下鼠标左键放置。



接着用鼠标点击接点的???处，并填入「Pump_Remote」的名称，输入时会出现智慧下拉选单，用户可以直接选择内容，也可以自行输入文字，输入完成后按下输入键。

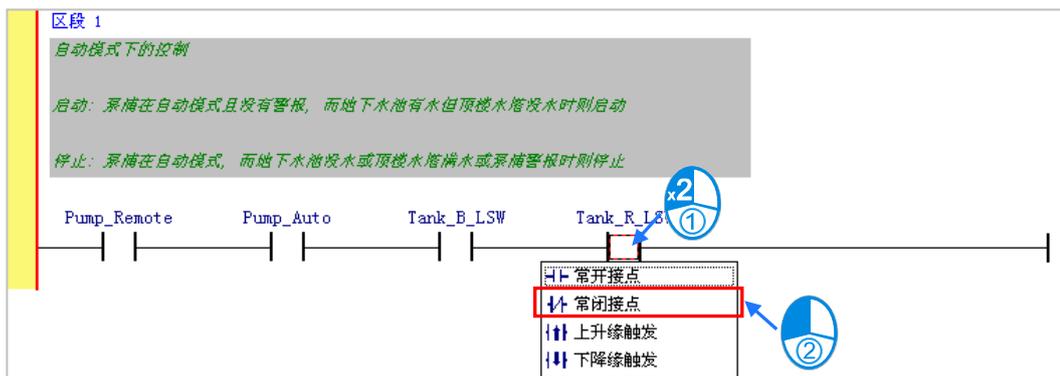
(输入文字后若出现地址的信息，则可以使用 按键来切换显示模式，但不管在哪种模式之下用户都可以文字方式或地址方式来编写程序内容)



利用同样操作完成以下程序内容。

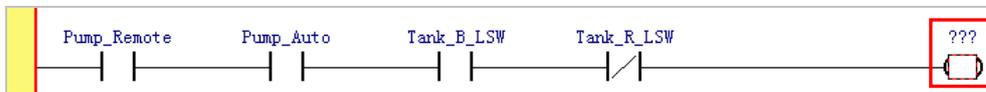
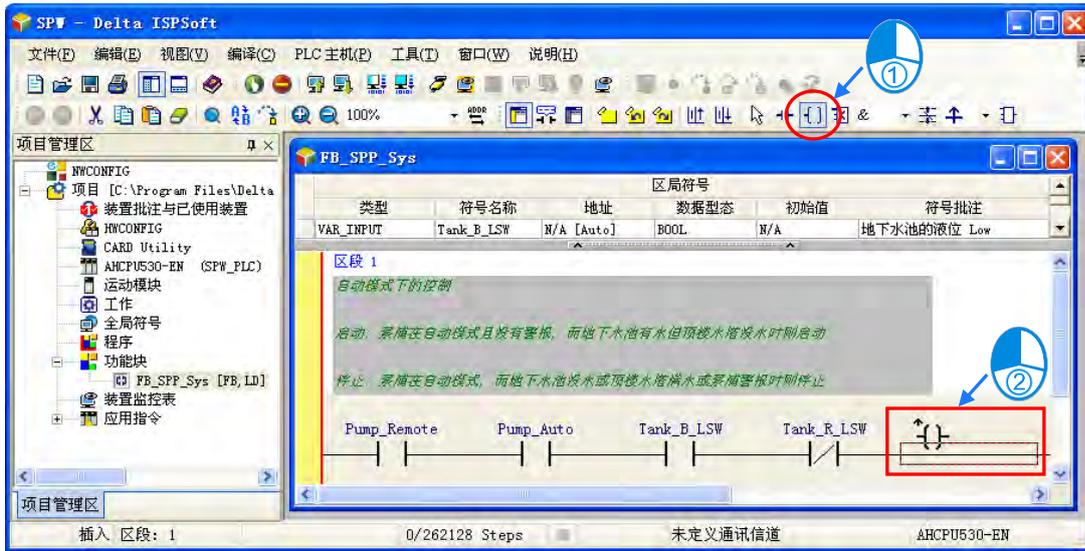


这时要地下水池有水而顶楼水塔没水的条件，所以 Tank_R_LSW 的状态必须为 False 才正确，这时可以使用鼠标连点两下 Tank_R_LSW 的接点图标，系统会显示接点选择的下拉选单，从里面选择我们需要的常闭接点型式。

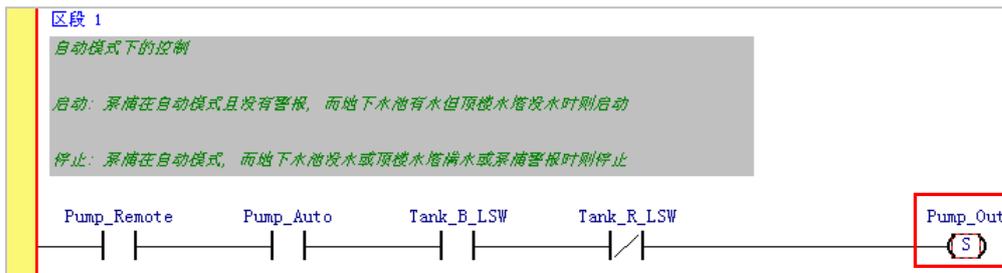


加入输出接点，一样在上方先选择  输出接点的图标，再利用鼠标在红框的位置点击放置。

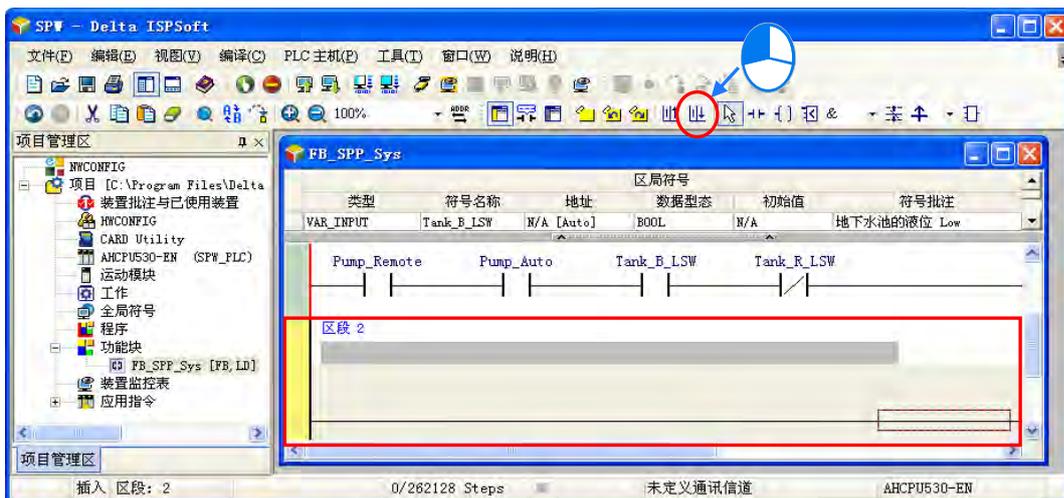
2



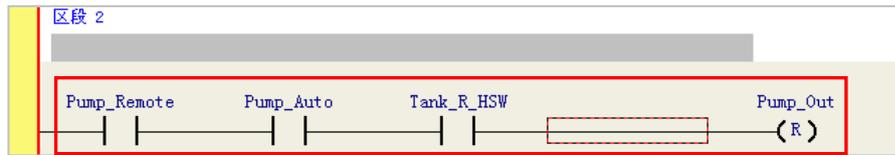
在接点的问号处填入「Pump_Out」，并依同样方式将输出接点设定成 Set 形式。



接着新增区段，先在上方找到  新增区段的图标，然后按下图标以便新增区段。

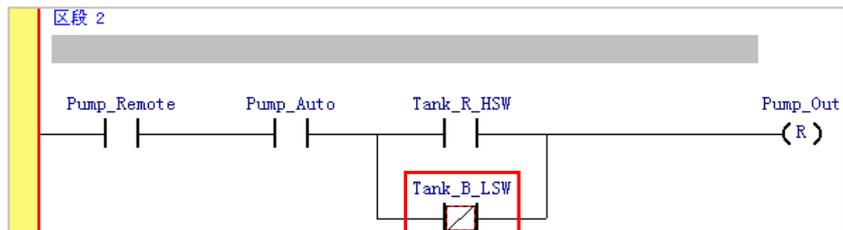
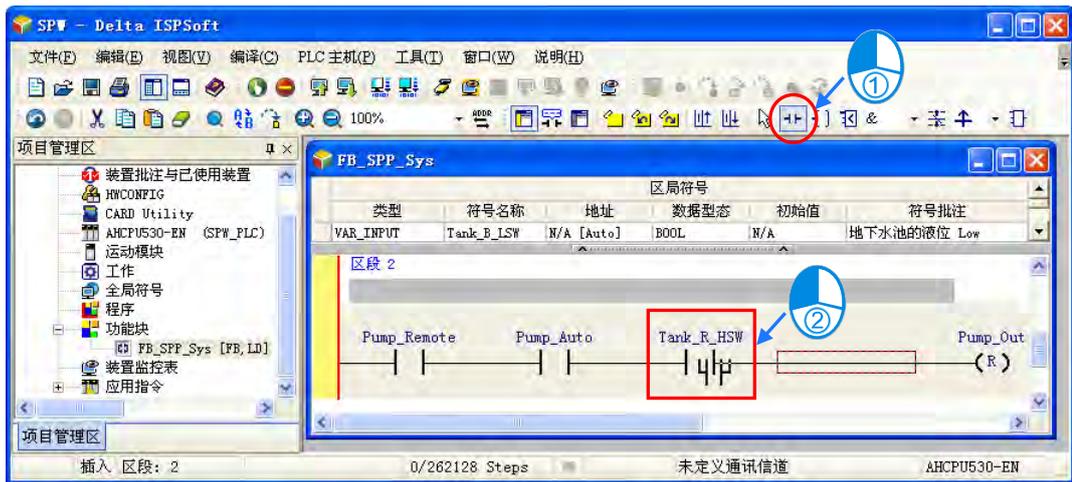


依之前的方式建立以下程序内容。

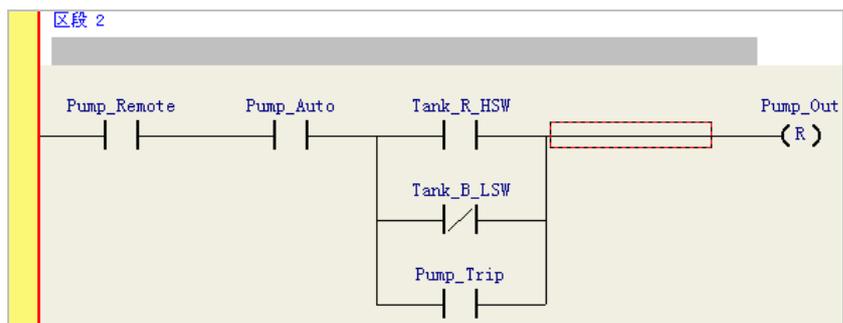


同时要考慮其他的停止條件，如地下水池沒水或泵浦發生故障，所以一樣在上方選擇 圖標，並將鼠標移到預並連的接點下方，當鼠標出現 的圖示時按下左鍵放置，然後在接點的???處填入「Tank_B_LSW」的名稱，接著改變接點的形式為常閉接點。

2

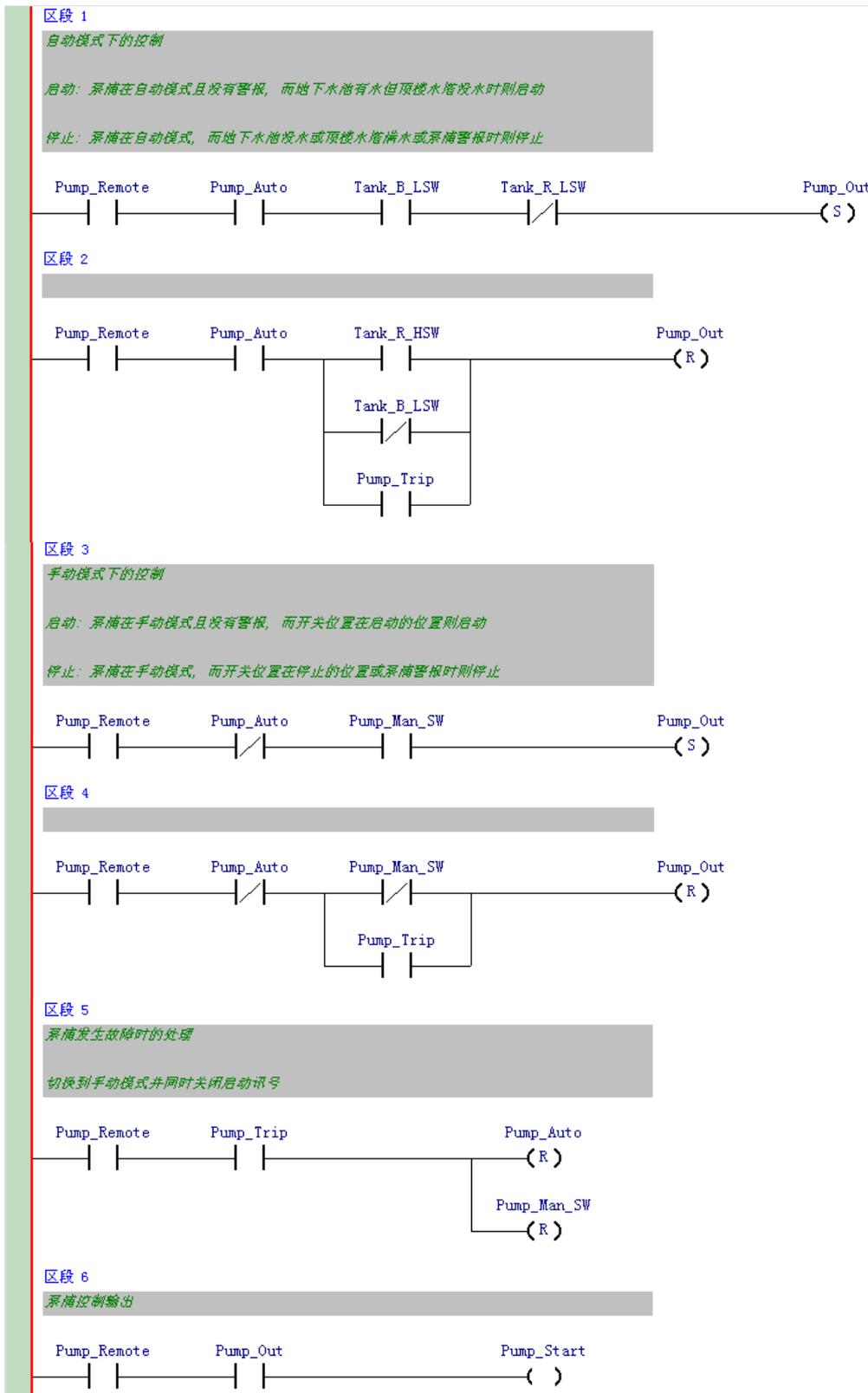


依同樣的操作方式完成以下程序內容。



这样就算完成自动的程控部份，接下来请依照同样的操作技巧与方式，完成以下的程序内容。

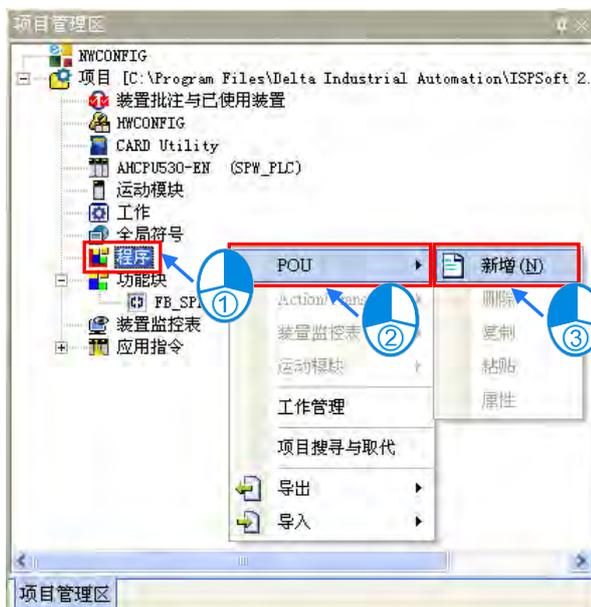
2



完成后存盘关闭这样就算完成了这个功能块 (FB) 的开发工作了。

2.9 建立主要程序

当功能块建立完成后就可以建立主程序了，首先在主画面左侧的窗口「项目管理区」内找到「程序」的图标与文字 ，利用鼠标右键的方式选择新增一个 POU。

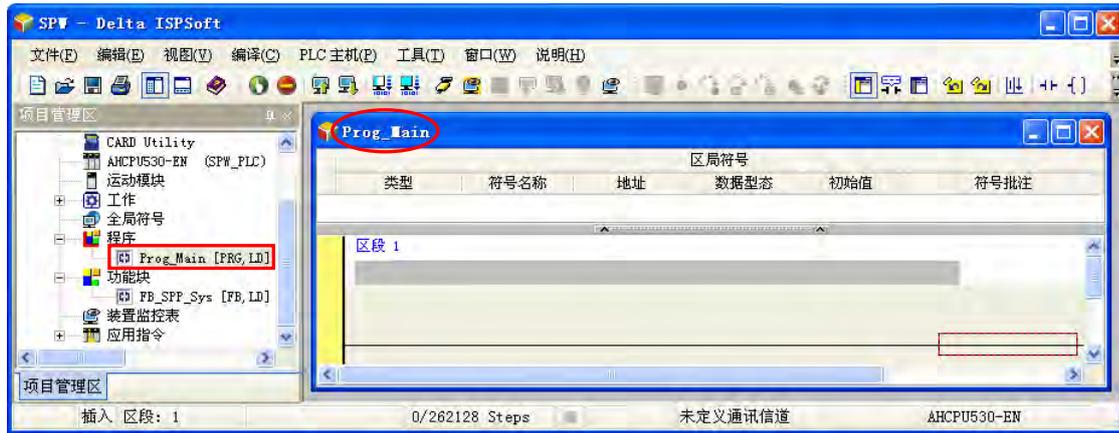


填入 POU 名称与注释，内容请参考下图，其余可保持默认值并按下确定。

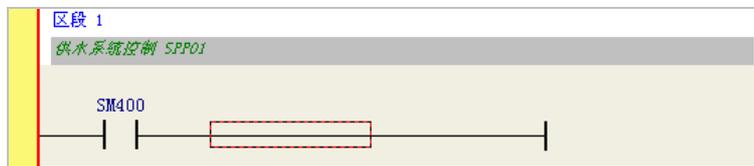


这样在主画面就可以看到主程序的程序编辑窗口。

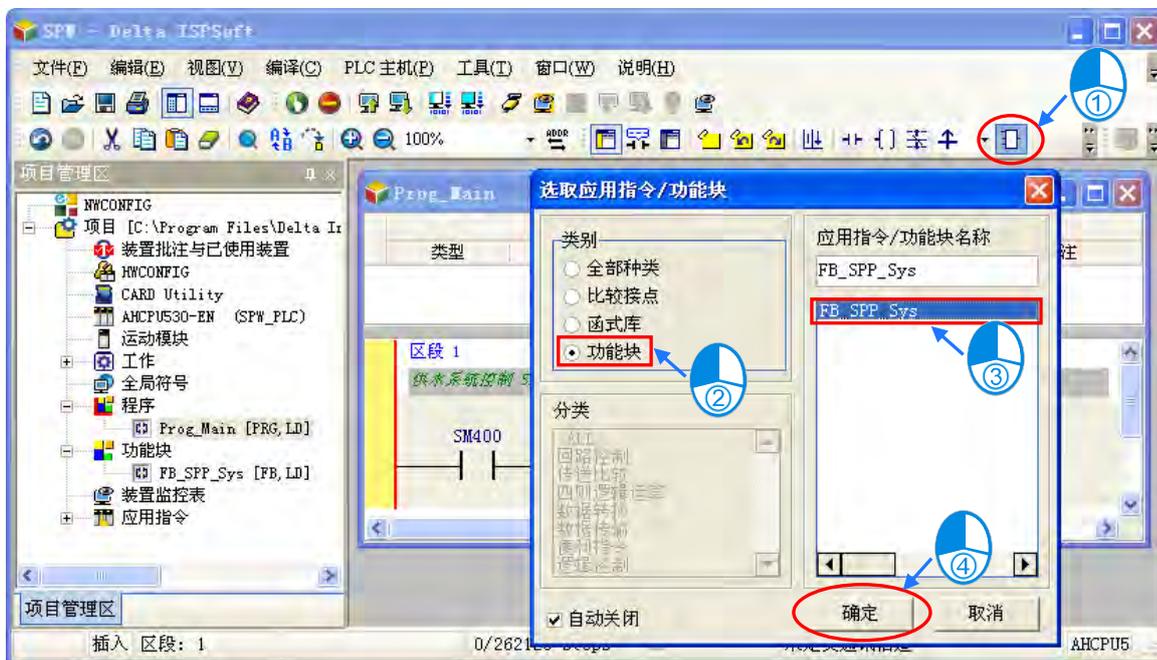
2



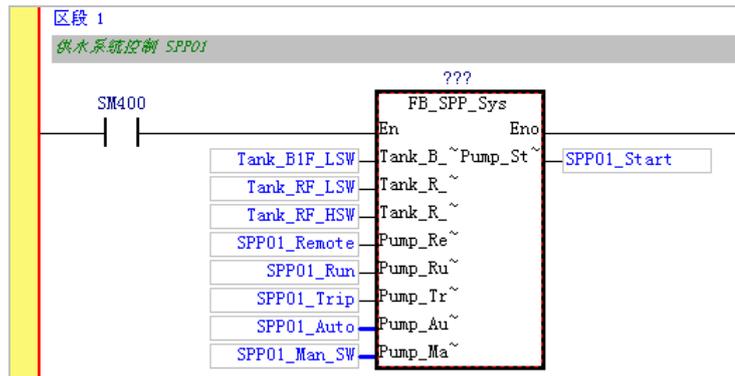
主程序的编辑环境与功能块一样，唯一的差别是功能块（FB）部份需要被调用才会执行，而程序（Program）部分是直接启动就会执行的，在本范例这里就不先规划主程序的区域符号了，所以请依照前面所学习的技巧，完成以下的程序内容。



接着准备调用功能块（FB）使用，一样先在上方找到调用功能块或应用指令的图标  并按下鼠标左键，接着就会跳出选择窗口，请在类别的地方选择功能块，并在功能块名称处选择「FB_SPP_Sys」，然后按下确定。



出现功能块的接口后，请依照下面的内容，将对应的全局符号名称填入到接脚内。

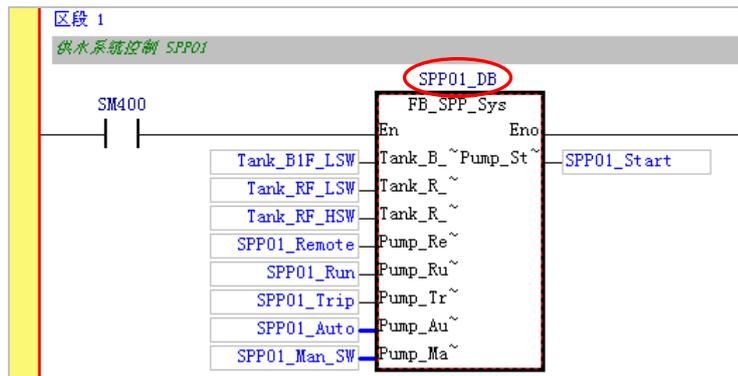


接着就准备建立一个数据区块给这个功能块使用，这个数据区块可以建立在全局符号或是区域符号，在这里我们把它建立在主程序的区域符号内，而建立的方式与建立一般的符号一样，只是在数据类型的部份要选择「功能块 (FB)」并指定「FB_SPP_Sys」的名称。请按照以下的内容建立一个「SPP01_DB」的符号名称。



当区域符号建立完成后，就可以在程序内容里的功能块之上，填入功能块类型的符号名称了。
(符号的数据类型必需要与功能块同名称才合法)

2

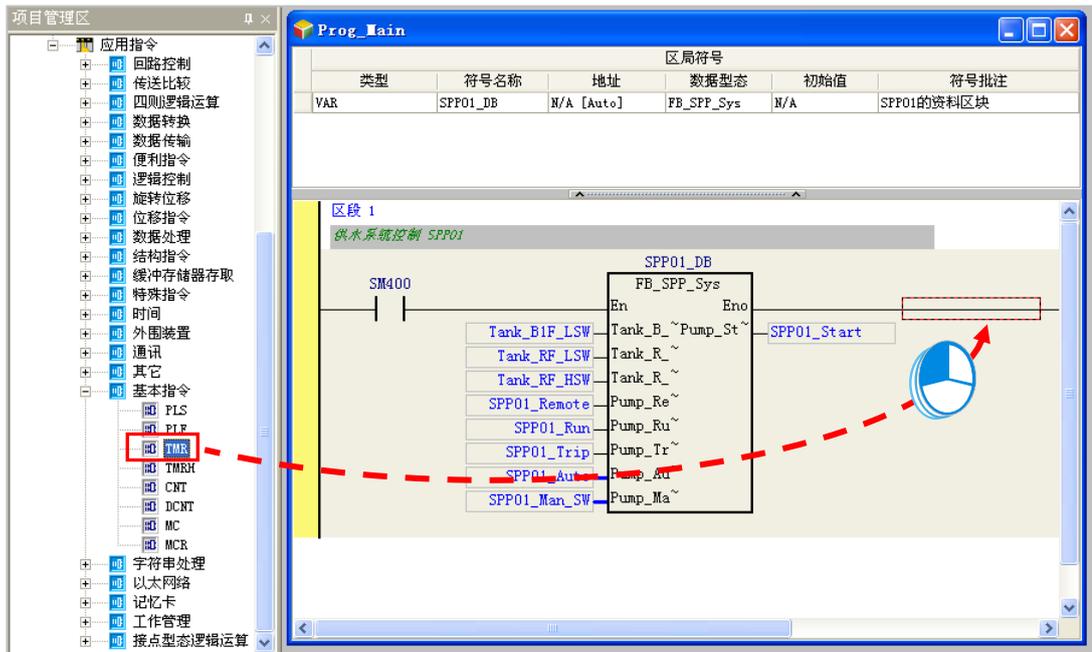


存盘后这样整个程序就算编写完成了。

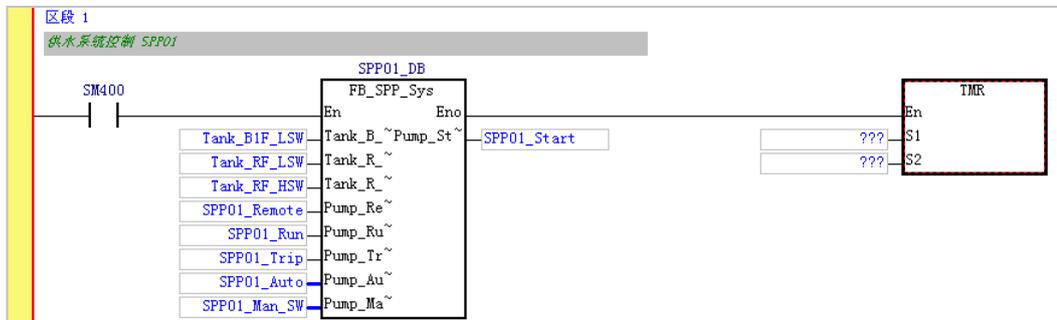
如果要测试目前开启的程序内容有没有语法上的问题，则可以使用检查语法的按钮  来进行语法测试，若要测试整个项目是否正确，则可以使用项目编译的按钮  来进行编译测试，项目编译的功能与检查语法不同，项目编译包含了检查语法的功能同时编译范围包含所有的程序与功能块内容。



注：本范例的内容为了让用户容易了解，所以以较简易的程序内容来展现，真的控制程序内容其实是比较复杂的，因为还可能会加入启动失败的 Fail 状态判断功能、Local 转 Remote 时的状态保持功能、运转时间的记录功能...等，而这些功能可能不是基本指令可以完成的，这时就可以拖曳主窗口左侧窗口内的应用指令  来帮助程序的编写 (详细内容可参考 ISPSoft 使用手册)。

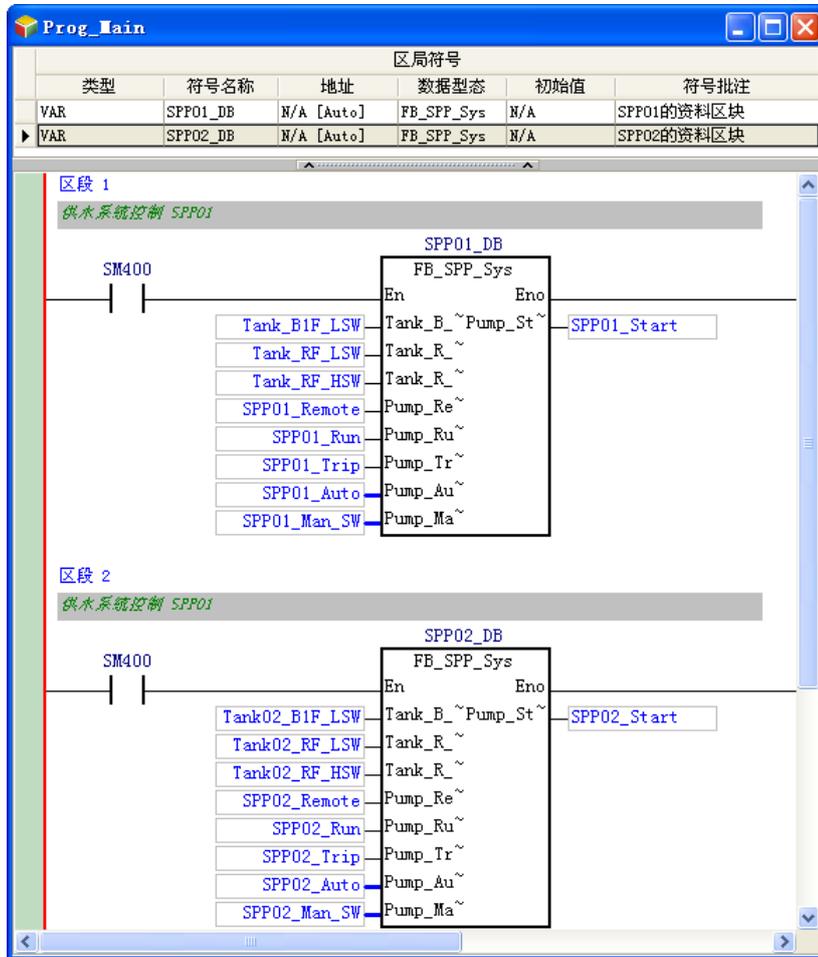


2



另外，若有第二套系统要建立时，则只要再调用一次「FB_SPP_Sys」，并把第二套的全局符号名称对应填入，即完成整个程序的编写，马上可以体会到功能块的便利性，但请注意在数据区块的使用上请勿共享，除非用户认为功能块内 VAR_OUTPUT 与 VAR 的区域符号在数值共享上没有影响，否则请另外建立一个功能块类型的符号给第二套系统使用，如下图就是分开建立的应用。

2



3

第3章 程序下载与监视

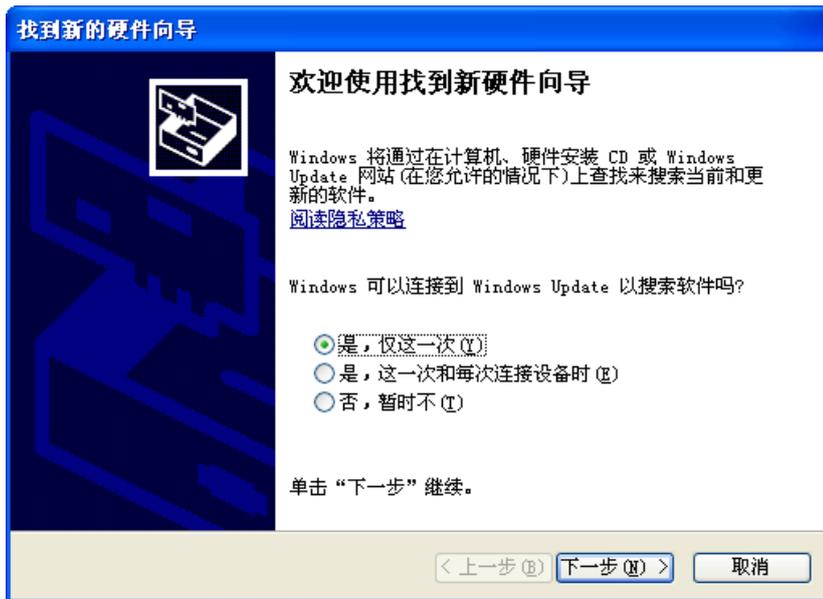
目录

3.1	COMMGR 设定	3-5
3.1.1	启动 COMMGR	3-5
3.1.2	开启 COMMGR	3-5
3.1.3	设定 COMMGR	3-6
3.2	专案下载	3-7
3.2.1	设定项目通讯	3-7
3.2.2	下载硬件配置	3-8
3.2.3	下载程序内容	3-9
3.3	程序监视与除错	3-10
3.3.1	程序监视	3-10
3.3.2	程序与系统除错	3-15

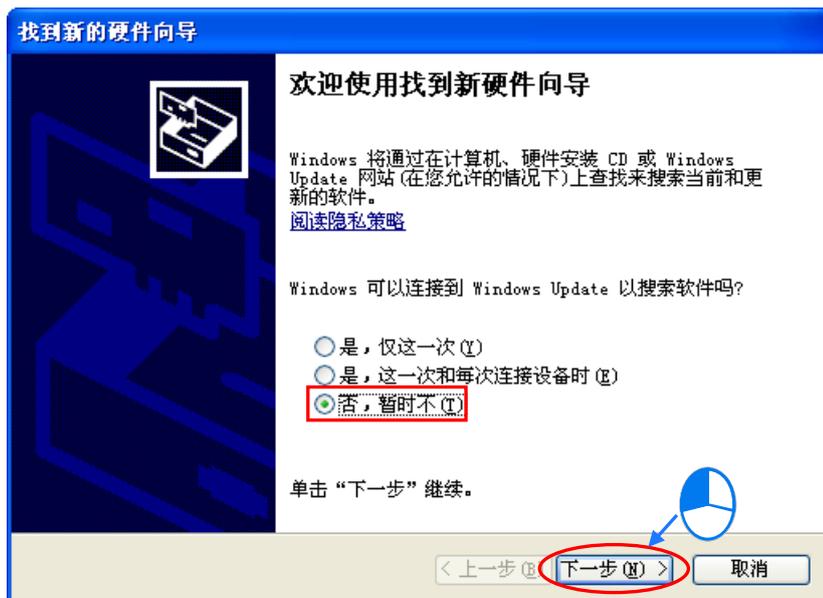
当程序编写完成后就可以准备进行程序下载的工作，因为最新版的软件套件已将通讯管理 (COMMGR) 与程序编辑 (ISPSOft) 拆分为两个主要软件以便功能与便利性的提升，所以除了准备好之前的硬件并上电以及安装主机的 USB 驱动程序之外，还要进行通讯驱动的软件设定才能与主机通讯。

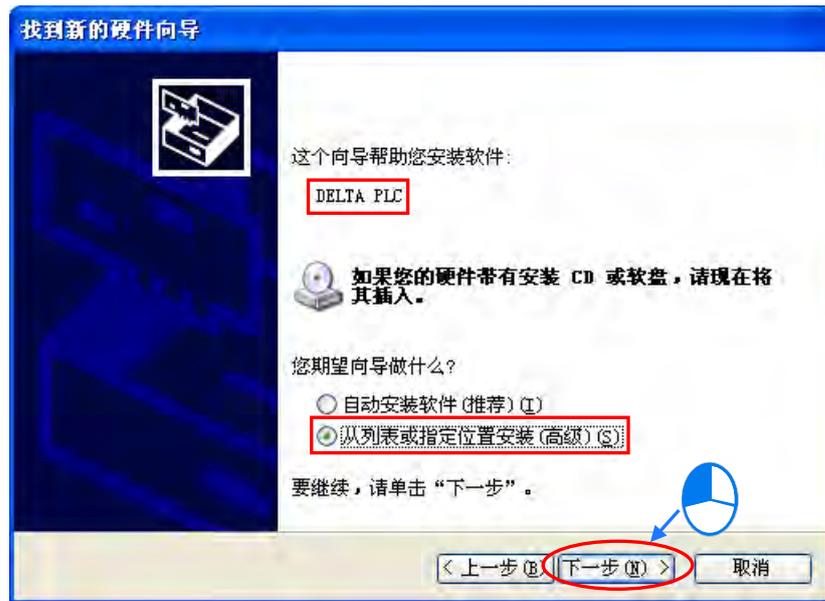
若计算机还没与主机的 USB 联机过，则请先将主机与计算机利用 USB 连接线的方式连接起来，这时会出现要求安装驱动程序的窗口画面。

3

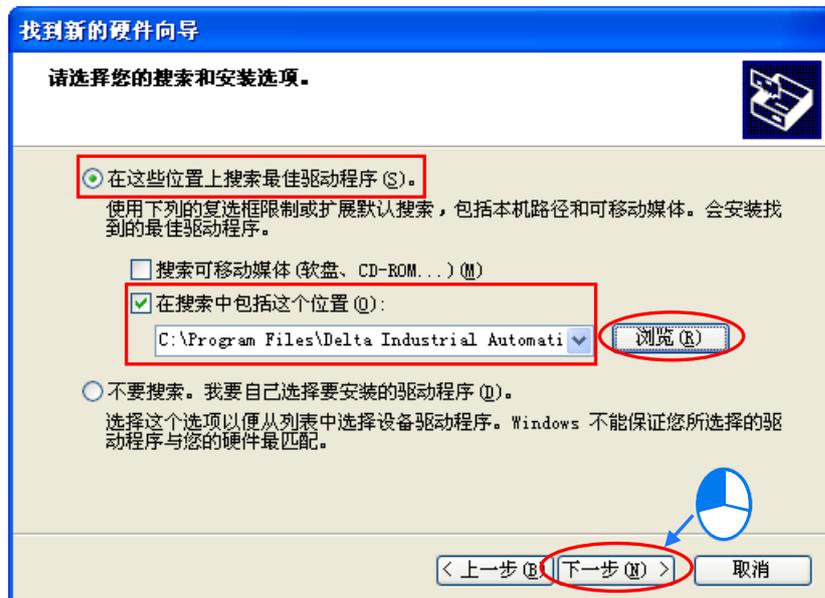


接下来请依以下内容进行操作。





以下图标内的路径为软件默认的安装位置「C:\Program Files\Delta Industrial Automation\ISPSoftx.xx\drivers\Delta_PLC_USB_Driver」, 若用户有修改过软件安装位置, 则请选择用户自定义的软件安装位置路径。



3



完成后就可以和设备管理器看到新增的 USB 驱动装置组件。(每个计算机与 USB 接口所配置的 COM 数值可能会不一样)



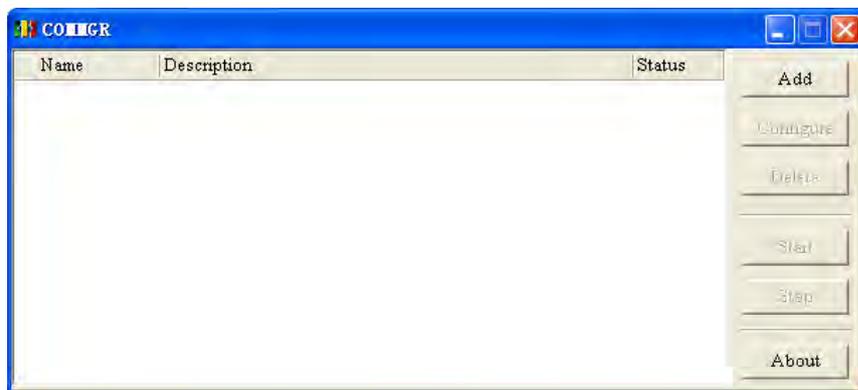
3.1 COMMGR 设定

3.1.1 启动 COMMGR

一般来说，软件安装完后 COMMGR 的程序快捷方式会自动放入操作系统的「启动」目录内，所以每次计算机开机后默认都是自动启动 COMMGR，并于操作系统的右下角任务栏内可以看到  的软件图标，若没有看到这样的图示，亦可以手动启动该软件，软件的启动快捷方式为「开始→程序集→Delta Industrial Automation→Communication→COMMGR→COMMGR」。

3.1.2 开启 COMMGR

启动后就可以开启 COMMGR 的接口以便设定，开启的方式为直接利用鼠标左键连点右下角的  图示两下，就可以开启如下图的设定接口。

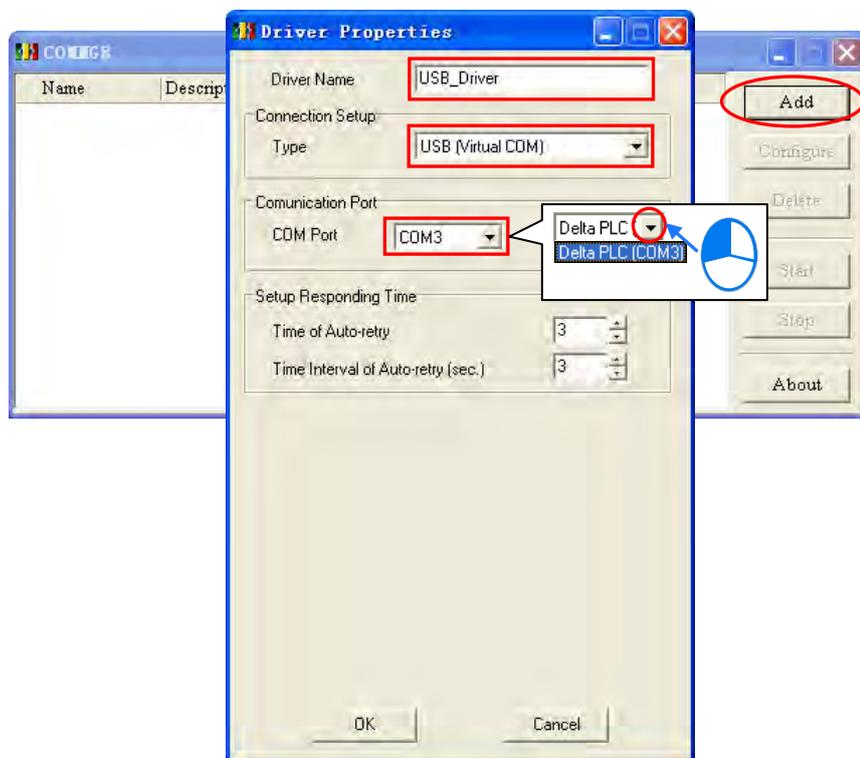


3

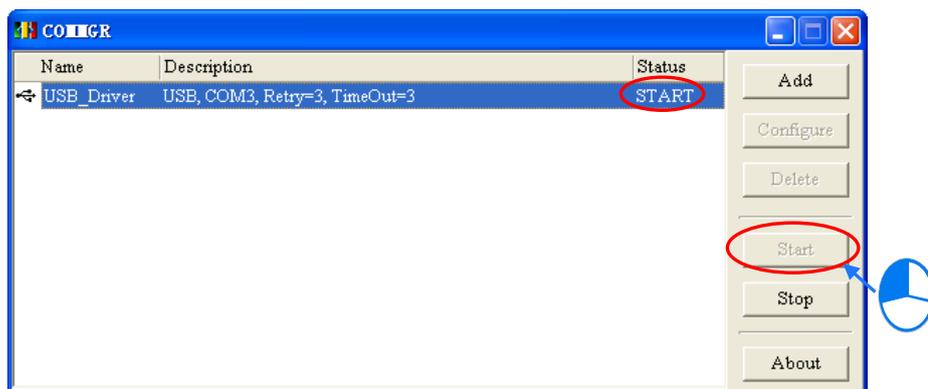
3.1.3 设定COMMGR

接口开启后就可以开始建立通讯的驱动设定，在这里我们选用 USB 连接线的方式来与主机做联机，所以会建立一个 USB 的驱动设定内容，先按下「Add」的按键就会出现一个设定窗口，然后依照下图的内容进行设定。

3



设定完成后就会多了一个驱动设定内容在窗口内，但在使用前必须要先将该驱动设定启动起来，所以请利用「Start」的按键将驱动设定启动起来。



3.2 专案下载

在完成上述的编辑之后，接下来便要將所编辑的参数及程序下载至主机中，而在本范例中需下载至 PLC 的数据有硬件架构、CPU 参数及项目程序。

3.2.1 设定项目通讯

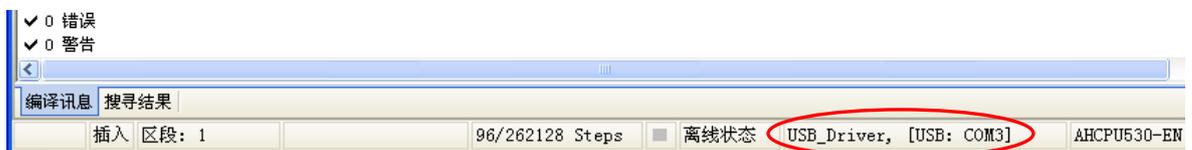
当 COMMGR 设定完成后，就可以在编辑软件 ISPSOft 的项目环境内设定通讯，首先在功能列找到「工具→通讯设定」。



出现以下窗口后，请选择之前设定的「USB_Driver」，按下确定即完成。



同时可以在画面右下角的状态栏看到这部份的设定信息。



3.2.2 下载硬件配置

首先 我们先下载硬件配置的部份 依照之前所教的方式打开 HWCONFIG 的硬件规划接口 而 ISPSOft 的通信设置会直接带入到 HWCONFIG 内。

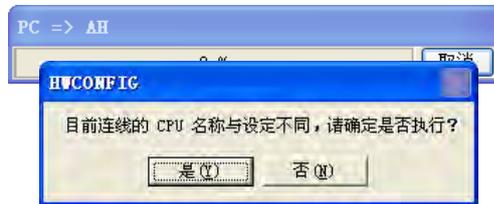
3



接着请按下  的图示，此时画面将会出现下载项目的提示窗口，按下确定之后，包含硬件架构以及 CPU 参数便会一并下载至主机中。



若实际主机内部的名称与 CPU 参数中的设定不同，此时系统会先弹出警示窗口，请确认实际联机的机器与我们想要下载的对象一致后，按下「是」开始进行下载。而下载完成后，主机原本有的错误灯号应该就会消失。



下载完成后关掉 HWCONFIG 接口窗口就完成了本范例的硬件下载工作。

(在实际应用上，若有使用到一些通讯或运动控制类型的模块时，也必须要有个别开启对应的软件，下载其专用的设定数据，才算是完成整个硬件规划的下载)

3

3.2.3 下载程序内容

硬件配置下载完成后就可以准备程序的下载部分，之前我们已经做过程序的项目编译部份，若用户有变动过程序的话，可以再按一次项目编译的按键来进行程序的检查，当编译成功之后就可以按下工具栏的下载图示  来进行程序的下载。



当传输设定的窗口出现后就可以进行下载内容的选择了，「执行码」的部份为主机运作必要的条件，所以是必须选择的选项，而在这个范例里面我们为了程序维护的方便，也会顺便勾选「程序」与「批注」的选项，这样系统会把目前的项目数据也备份一份在主机内，以避免程序的遗失，或用户到现场修改程序却忘了带原始开发程序时，也可以利用上传的功能，将原始开发程序从主机内提取出来以便修改。

3



接下来再按下确定与传送就完成了程序的下载操作了。



最后检查一下模块的 I/O 灯号与现场设备状态，确认整个系统是可以进行系统测试的阶段，即可将主机的切换开关由 STOP→RUN 进行系统测试。

3.3 程序监视与除错

3.3.1 程序监视

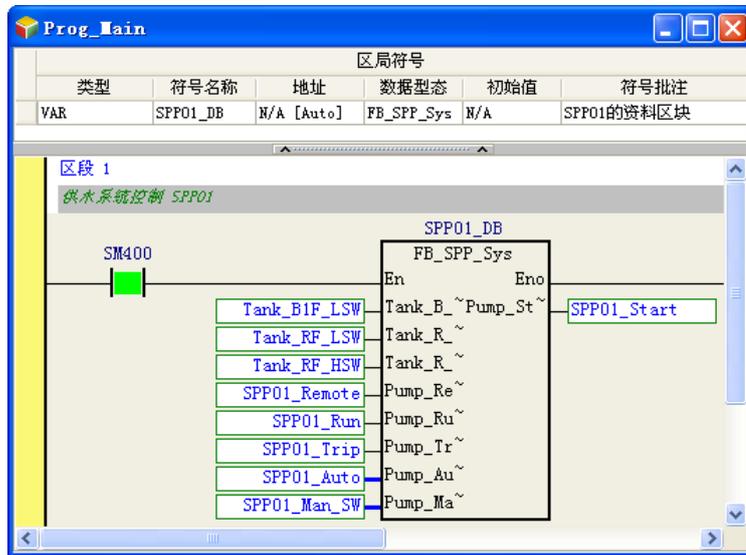
当程序运行的时候，用户可以通过监视的方式，来了解系统目前的逻辑控制状态，或是修改部分装置的数值来进行系统的测试，现在我们就来进行程序监视的操作。

- 由程序内容监视

打开之前我们所编写的主程序「Prog_Main」，然后按下联机的图示 。

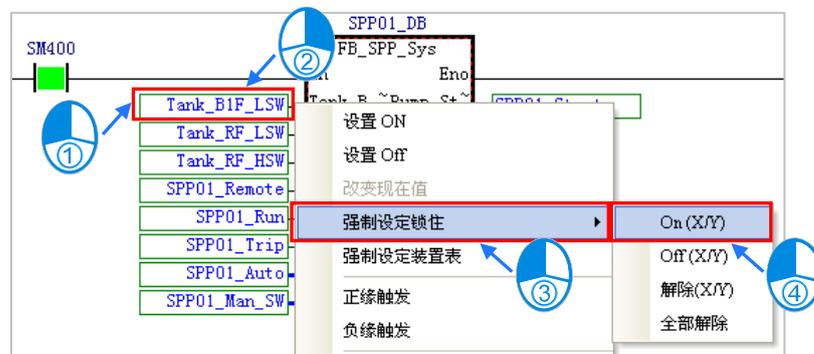


接着画面就会进入监视状态，程序内容除了本身的逻辑程序之外，也包含了所有装置的目前信息。（BOOL 类型会以绿底（ON）或白底（OFF）来表示，其他类型会直接显示数字或文字）

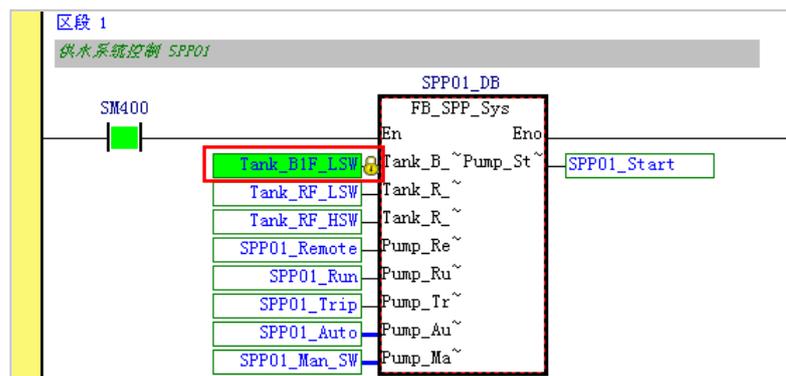


3

若想要修改装置的内容，可以在想要修改的装置上，使用鼠标右键的功能选单来进行修改，在这里我们想进行修改「Tank_B1F_LSW」的装置，要注意的是该装置为对应到实体 I/O 的装置，所以若使用一般的 ON/OFF 修改是无法变动的，因为实体的 I/O 数值会马上覆盖掉我们送出的修改值，所以我们会使用强制 ON/OFF 的操作来修改该装置数值。



装置数值会被修改，同时会出现锁头的符号，来表示该装置为强制的状态。



测试完之后要记得取消强制的状态让主机恢复正常。



3

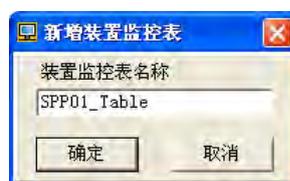
● 由装置监控表监视

有的时候在测试逻辑控制时，要修改的装置往往会散布在各个程序段落内，直接在程序内寻找监视并修改是很不便的一件事。另外，有的时候修改装置值并不是为了程序的除错，而是为了测试外部的装置而提供类似图控的修改数值功能，在这样的情况下，若使用程序监视方式来修改，一方面装置会散落在程序各处不易寻找，另一方面手上必须要有与主机相同的程序才可以。而为了解决这方面的问题，我们可以使用「装置监控表」的方式来进行监控，使用这类的方式来修改装置数值时，甚至不需要项目程序就可以进行监视与修改装置的工作。

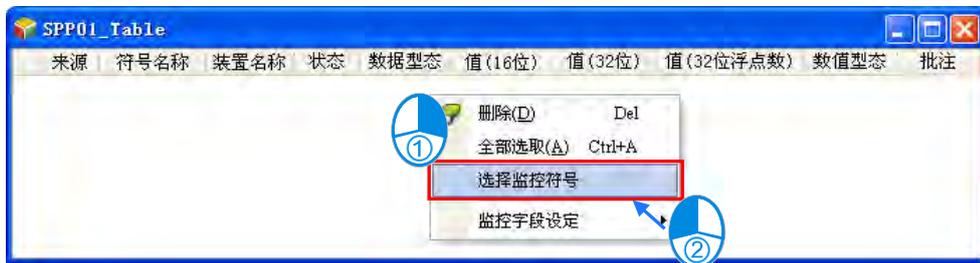
首先我们在左侧的窗口内，找到装置监控表的文字，并使用鼠标右键的功能选单来新增。



请输入设备监控表的名称「SPP01_Table」并按下确定。



与符号建立方式相同，利用连点空白处两下或鼠标右键的方式来新增内容，在这里我们使用鼠标右键的方式来建立。



按下全部选取后按确定带入符号。



符号带入后再按取消并存盘就算完成这次的装置监控表格了。

来源	符号名称	装置名称	状态	数据型态	值 (16位)	值 (32位)	值 (32位浮点数)	数值型态	批注
Global Symbols	Tank_B1F_LSW		<input type="checkbox"/>	BOOL					地下水池的液位
Global Symbols	Tank_RF_LSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	Tank_RF_HSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	SPP01_Remote		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Run		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Trip		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Start		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	Tank_B1F_LT		<input type="checkbox"/>	REAL				有号数十进	地下水池液位 L
Global Symbols	SPP01_Auto		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Man_SW		<input type="checkbox"/>	BOOL					地下供水 泵浦

接下来请按下装置监控的按钮 ，本按钮仅针对装置部份进行监视与修改，所以并不需要项目程序的副本即可进行操作。

3

SPP01_Table									
来源	符号名称	装置名称	状态	数据类型	值 (16位)	值 (32位)	值 (32位浮点数)	数值型态	批注
Global Symbols	Tank_B1F_LSW		<input type="checkbox"/>	BOOL					地下水池的液位
Global Symbols	Tank_RF_LSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	Tank_RF_HSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	SPP01_Remote		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Run		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Trip		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Start		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	Tank_B1F_LT		<input type="checkbox"/>	REAL	20972	100193534	0.006	有号数十进	地下水池液位 L
Global Symbols	SPP01_Auto		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Man_SW		<input type="checkbox"/>	BOOL					地下供水 泵浦

与程序监视相同，在装置上使用鼠标右键方式，就可以进行装置的修改操作，在这里我们修改内部装置「SPP01_Man_SW」的状态，因为不是对应实体I/O的装置，所以直接使用一般的ON/OFF即可。

SPP01_Table									
来源	符号名称	装置名称	状态	数据类型	值 (16位)	值 (32位)	值 (32位浮点数)	数值型态	批注
Global Symbols	Tank_B1F_LSW		<input type="checkbox"/>	BOOL					地下水池的液位
Global Symbols	Tank_RF_LSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	Tank_RF_HSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	SPP01_Remote		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Run		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Trip		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Start		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	Tank_B1F_LT		<input type="checkbox"/>	REAL	-15729	997573263	0.004	有号数十进	地下水池液位 L
Global Symbols	SPP01_Auto		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Man_SW		<input type="checkbox"/>	BOOL					地下供水 泵浦

设置 ON

设置 OFF

强制设定锁住

改变现在值

修改后如下图。

SPP01_Table									
来源	符号名称	装置名称	状态	数据类型	值 (16位)	值 (32位)	值 (32位浮点数)	数值型态	批注
Global Symbols	Tank_B1F_LSW		<input type="checkbox"/>	BOOL					地下水池的液位
Global Symbols	Tank_RF_LSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	Tank_RF_HSW		<input type="checkbox"/>	BOOL					顶楼水塔的液位
Global Symbols	SPP01_Remote		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Run		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Trip		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Start		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	Tank_B1F_LT		<input type="checkbox"/>	REAL	23593	999250985	0.004	有号数十进	地下水池液位 L
Global Symbols	SPP01_Auto		<input type="checkbox"/>	BOOL					地下供水 泵浦
Global Symbols	SPP01_Man_SW		<input checked="" type="checkbox"/>	BOOL					地下供水 泵浦

如此就完成了装置监控部分的操作，若不需要监视时则可点一下  图标，以离开监视模式然后关闭装置监控表即可。

3.3.2 程序与系统除错

当系统运行的时候，有时会有错误发生，有的错误会导致主机或模块的错误灯号亮起，有的错误并不会引起任何的灯号，后者的错误大都是逻辑性程序错误的情形较多，在这次的范例操作当中，若用户都依照程序来操作的话，则应当不会有发生错误的情形。为了本章节的示范操作，接下来会请使用者将电源供应器（24VDC）的电源关闭，以制造一个系统错误。当电源一关闭后，应该就会发现到主机的BUS FAULT灯号与06XA的ERROR指示灯号亮起。

接下来一样在左侧窗口找到HWCONFIG并开启，然后点击联机的图示 ，就会发现软件画面会直接显示实际模块的状态。



一般来说，主机的BUS FAULT灯号亮起都是指硬件架构有错误，所以我们可以选择一下目前有错误的06XA模块，然后利用右键的选单方式选择「诊断」。

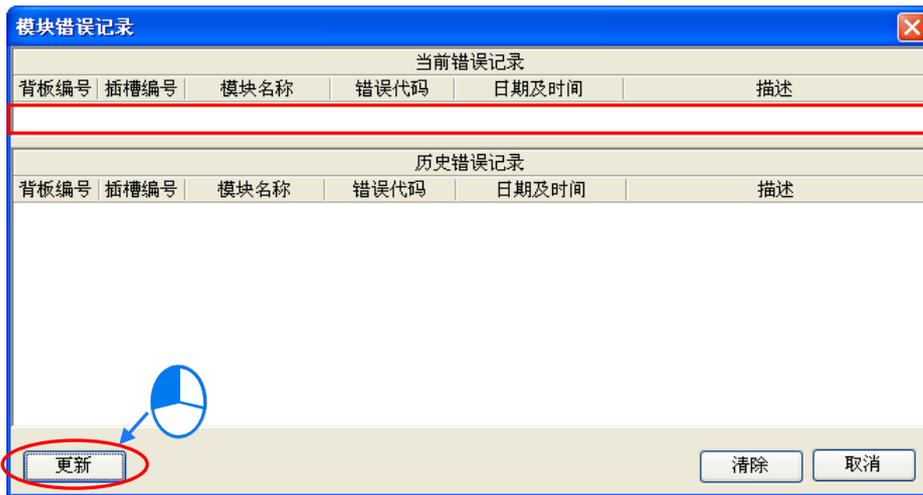


在下图我们可以看到错误的信息。



可以发现是该模块未供电，这当然是我们故意测试的，接着请把电源供应器的电源再恢复送上，然后按下「更新」的按钮，此时系统便会再次确认该模块的状态，而在本范例中，因模块电源已正常供应，因此系统将不会检测到错误的信息，而当前错误纪录的窗口中也会呈现空白（如下图）；当然，若系统有检测到新的错误产生时，新的错误信息便会重新显示在「当前错误纪录」的窗口中；另外，按下「清除」钮则可将下方的历史错误纪录清除，不过依据各模块设计的不同，并非所有模块都会保留错误纪录，因此，其历史错误纪录的字段便可能一直是空白的状态。

3



然后关闭该窗口回到系统架构主画面，也会发现整个系统都恢复正常了。



这时再利用主机的RUN/STOP切换开关，将开关移动到STOP的位置后再移到RUN位置，主机启动后系统与软件画面都会回复到原本正常状态。



关闭硬件规划窗口回到ISPSOFT主画面，找到菜单单的「PLC主机」→「PLC错误纪录」，也可以显示刚刚的错误信息。



由表格中可以看到模組的電源異常錯誤。



背板编号	插槽编号	模块 ID	错误代码	日期 & 时间	描述
1	-	AHCPU530-EN	16#1801	0-1-1 2:28:47	=CPU模块未设定中
1	2	AH06XA-5A	16#A601	0-1-1 2:20:23	电源异常
1	2	AH06XA-5A	16#A601	0-1-1 1:3:19	电源异常
1	-	AHCPU530-EN	16#1801	0-1-1 1:3:17	=CPU模块未设定中
1	2	AH06XA-5A	16#1402	0-1-1 1:1:14	IO模块不符合IO配置
1	2	AH06XA-5A	16#1402	0-1-1 0:59:54	IO模块不符合IO配置
1	-	AHCPU530-EN	16#1801	0-1-1 0:0:6	=CPU模块未设定中
-	-	AHCPU530-EN	16#1400	0-1-1 0:0:1	协处理器存取错误

若不需要旧的记录信息，也可以按下「清除记录」的按键以便阅读的方便性，然后关闭所有软件就算完成了这次的范例操作。

3

MEMO

3