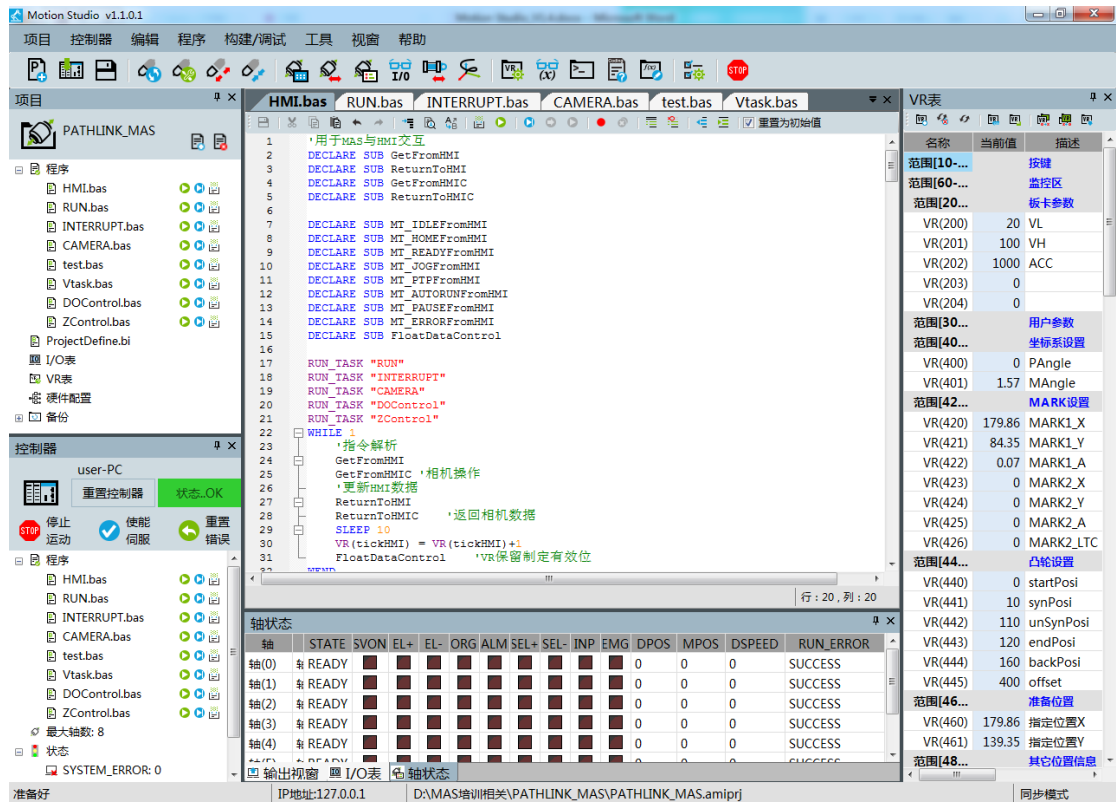


# 研華MAS控制器 Motion Studio手冊

Version 1.6

## 1. 簡要介紹

Motion Studio 是基於 Windows 平臺下的一個應用程式，用於研華 MAS 控制器的程式開發和調試。Motion Studio 提供使用者最簡單的方式去設定控制器，快速開發並且能夠線上監控(run-time diagnostics) 程式在控制器中的執行狀態。



### 1.1 系統要求

運行 Motion Studio 軟體的電腦配置需求如下

	最低	推薦
作業系統	Window XP SP3	Windows 7 Windows 8 Windows 10
.NET Library	.NET 4.0 及以上	
處理器		
RAM	2M	4M
硬碟空間	150M	300M

### 1.2 操作模式

Motion Studio 程式設計環境有 3 種操作模式

- 斷線模式 (Disconnected Mode)  
Motion Studio 跟控制器未連線，所有的功能都無法使用，通訊連線為關閉狀態。
- 監控模式 (Tool Mode)


選擇監控模式連線，**Motion Studio** 可使用所有的 **Tools** 來監控控制器的狀態與所有軸與 I/O 的狀態，也可以編譯/運行程式或停止程式的運行，但不可以在編輯器中編輯程式。

- 同步模式 (Sync Mode)

在此模式使用者可以做任何的操作，包含編寫程式,下載程式,執行程式和用測試工具測試每個軸，每個 IO 點等。

## 2. 開發步驟

### 2.1 與控制器建立連接

初次打開 Studio 後，若 Studio 和控制器不是同一台電腦，兩者沒有建立連接，需要先與控制器建立連接。點擊主功能表的控制器—>連接設置或者點擊工具按鈕，將彈出以下對話方塊：



The dialog box is titled "连接配置" (Connection Configuration). It has two main sections: "接口" (Interface) and "连接参数" (Connection Parameters). In the "接口" section, the "以太网" (Ethernet) radio button is selected, and the "模拟器" (Simulator) radio button is unselected. In the "连接参数" section, the "IP地址" (IP Address) field contains "127.0.0.1" and the "IP端口" (IP Port) field contains "6000". At the bottom, there are two buttons: "应用" (Apply) and "取消" (Cancel).

當選擇介面為乙太網時，輸入控制器的 IP 位址和 IP 埠，點擊“應用”，與控制器建立連接。此時，表示使用實際軸卡。

當選擇介面為模擬器時，可選擇與 MAS-324X Simulator（4 軸虛擬控制器）或者 MAS-328X Simulator（8 軸虛擬控制器）進行連接。



The dialog box is titled "连接配置" (Connection Configuration). It has two main sections: "接口" (Interface) and "连接参数" (Connection Parameters). In the "接口" section, the "以太网" (Ethernet) radio button is unselected, and the "模拟器" (Simulator) radio button is selected. In the "连接参数" section, the "MAS-324X Simulator" radio button is selected, and the "MAS-328X Simulator" radio button is unselected. At the bottom, there are two buttons: "应用" (Apply) and "取消" (Cancel).

#### 2.1.1 切換控制器選擇配置來源

切換控制器，考慮到不同控制器配置可能不同，使用者可能希望採用的配置也不一樣，因此

目前有提供三種方式供使用者選擇，如下面的對話方塊：



目前的連接方式可分為以下三種：

- 1) Ethernet，即與實體控制器連接
- 2) MAS-324X Simulator：與虛擬控制器進行連接，虛擬控制器型號：MAS-324X Simulator
- 3) MAS-328X Simulator：與虛擬控制器進行連接，虛擬控制器型號：MAS-328X Simulator

舉例說明：

用戶在筆記型電腦上與虛擬控制器 MAS-328X Simulator 中連接，調試好程式後，到實機上進行測試會改為連接到 Ethernet，進行相應配置。後面又需要在筆記型電腦上偵錯工具，切換回 MAS-328X Simulator。

- 導入配置

每次關閉或切換連接時，會保存對應連接方式的配置。

選擇【導入...】，將使用對應連接方式最後一次的配置。

對應上面的例子，再次切換到 MAS-328X Simulator 時，選擇【導入...】，則會導入之前的 MAS-328X Simulator 的相關配置。

- 複製配置

選擇【複製...】，會複製上一次連接方式的配置。

對應上面的例子，再次切換到 MAS-328X Simulator 時，選擇【複製...】，則會複製實機上配置到當前的 MAS-328X Simulator。

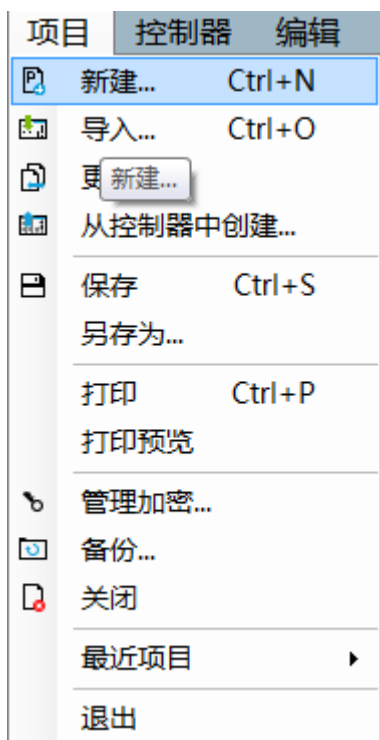
- 重新產生

選擇重新產生或者關閉此對話方塊，都會按照目前連接方式的可用板卡產生預設配置。

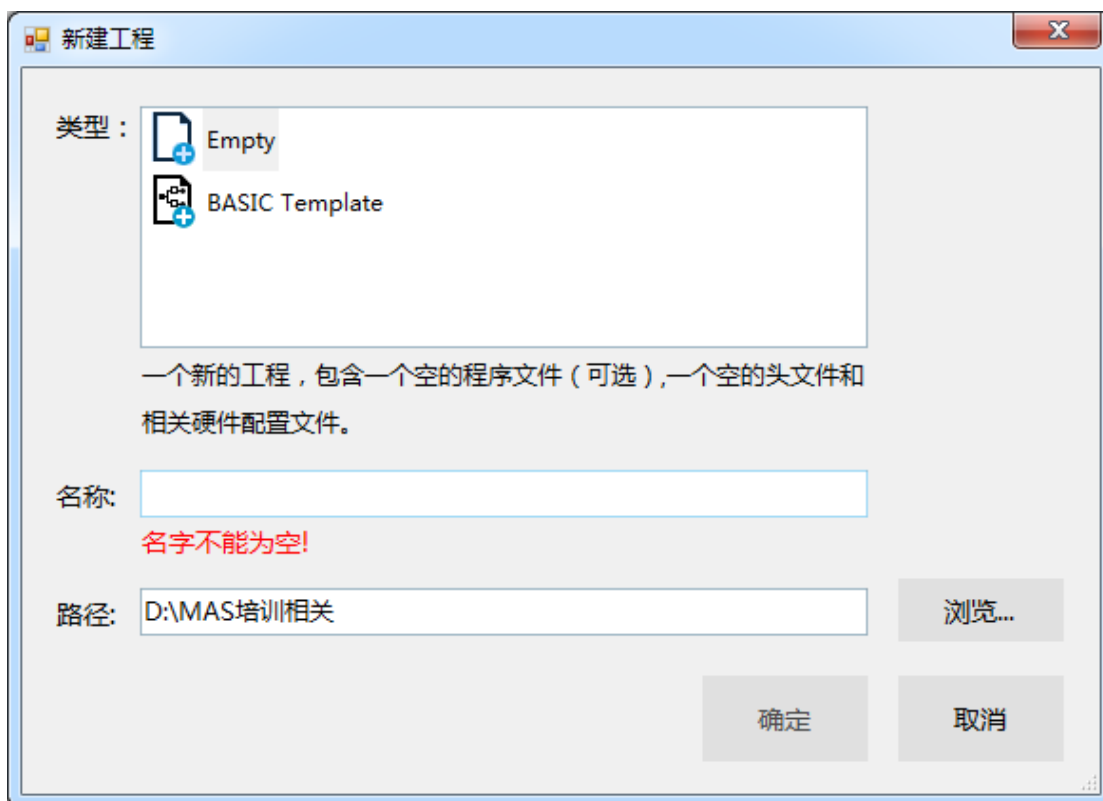
對應上面的例子，再次切換到 MAS-328X Simulator 時，選擇【重新產生（默認）】，則會產生 MAS-328X Simulator 的預設配置（注：預設的配置沒有氣油缸相關配置）。

## 2.2 新建項目

點擊主菜單——>專案——>新建...，創建新的專案



選擇專案類型（預設為 Empty），選擇工程要保存的路徑後，輸入工程名（工程名只能是字母/數位/底線），點擊“確定”或者按 Enter 鍵將創建一個新的工程。

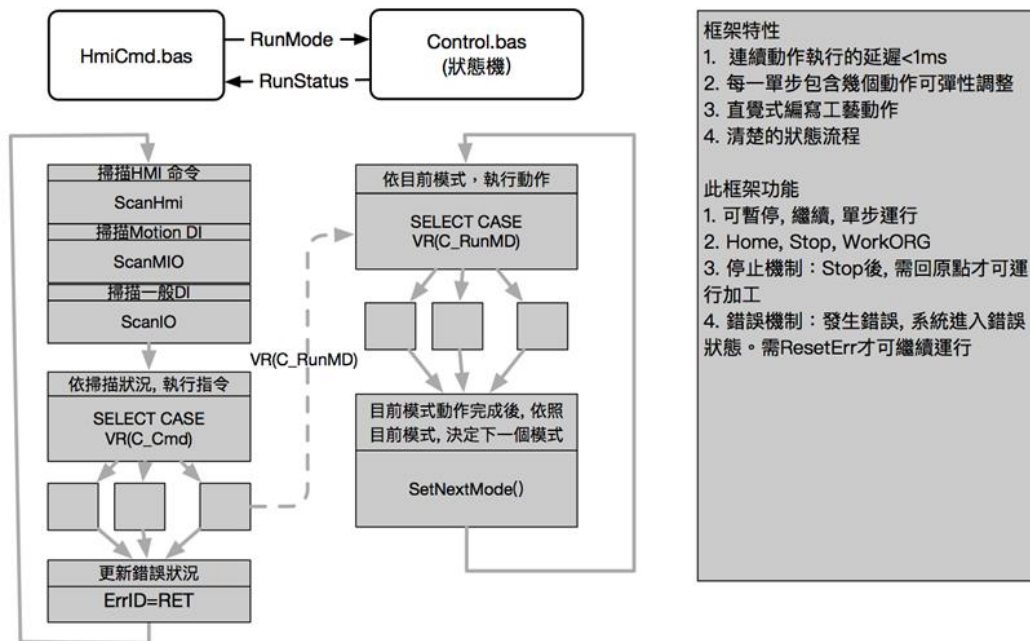


專案類型說明：

Empty：新的工程，包含一個空的.bas 檔，一個空的標頭檔 ProjectDefine.bi，同時會創建目前可用的硬體設定檔。

BASIC Template：新建一個範本工程，包含兩個.bas 文件，一個頭文件 ProjectDefine.bi，同時會創建目前可用的硬體設定檔。其中.bas 和 ProjectDefine.bi 中提供開發專案的常用框架（具體介紹請參考說明——>範本框架使用者手冊）。

## MS Basic模版框架

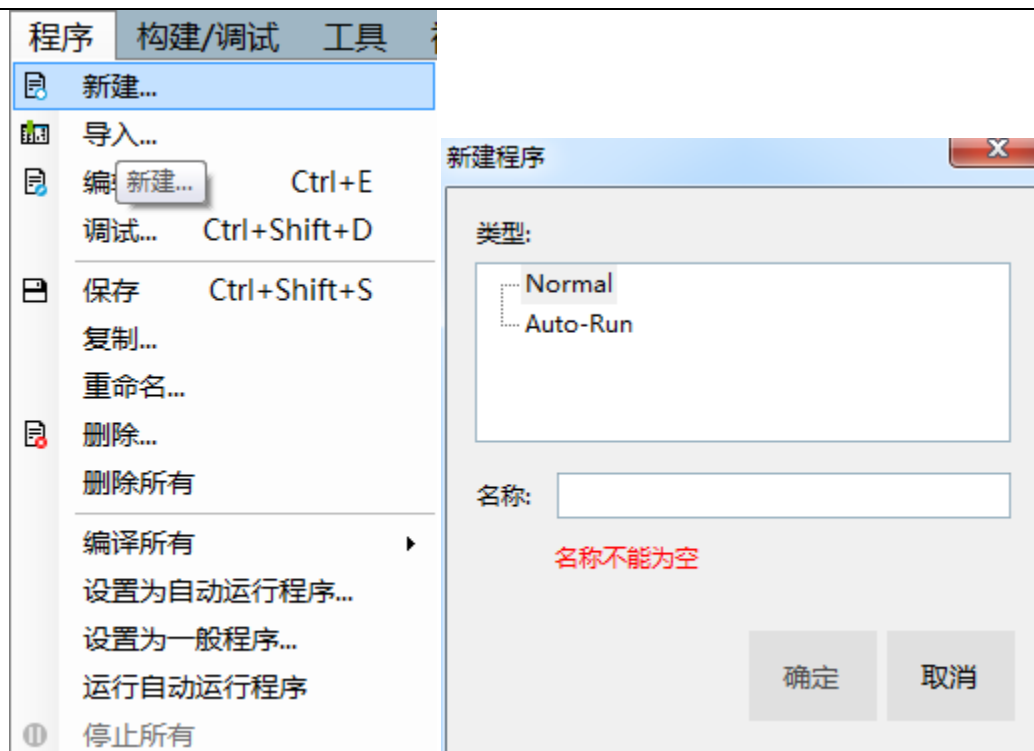


注：創建專案會同時創建一個與專案名相同的資料夾，資料夾的名稱一定要與專案工程的名稱一致，否則 **Controller** 將不識別。因此，請切記在 **Studio** 以外私自更改檔（資料夾）的名稱/內容！

也可通過專案——>導入...，切換到另外一個已存在的項目。

## 2.3 新建程式

點擊主功能表——>程式——>新建...，創建新的程式。在跳出的表單中選擇程式類型（預設為 Normal），點擊“確定”或者按 **Enter** 鍵將創建新的程式。

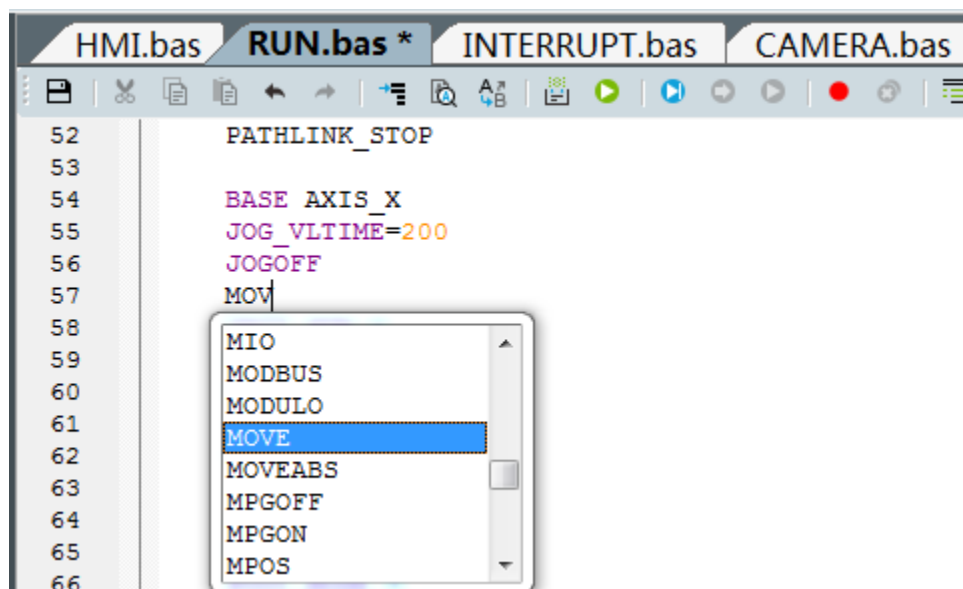


也可通過程式—>導入...，導入其他專案中已存在的程式

## 2.4 編寫 BASIC 代碼

新建程式後即可在編輯方塊中編輯 BASIC 指令。可使用上面的工具按鈕進行相應編輯，如後面四個按鈕依次含義為：注釋、取消注釋、減少縮進和增加縮進。

勾選“重置為初始值”，則在此.bas 運行前會將 VR 表的 VR 變數和 IO 表的 IO 變數重新設置為對應的初始值。



## 2.5 編譯/調試/運行

編寫完 BASIC 後，可點擊各個編輯方塊的工具按鈕進行編譯/調試/運行。

主功能表和編輯方塊中編譯/調試/運行相關工具按鈕的含義分別為：



- ：編譯
- ：發佈版運行
- ：調試版運行
- ：單步
- ：停止運行

## 2.6 工具

- 軸運動測試：軸測試

軸測試工具：可進行 PTP/JOG/Homing 運動測試



- 查看軸屬性：軸參數

可查看配置後的指定軸的屬性。

轴参数			
参数	轴(0)	轴(1)	轴(2)
UNIT_NUM	1	1	
UNIT_DENO...	1	1	
Speed			
MAXVEL	5,000,000	5,000,000	5,000,000
MAXACC	50,000,000	50,000,000	50,000,000
MAXDEC	50,000,000	50,000,000	50,000,000
VH	8,000	8,000	8,000
VL	2,000	2,000	2,000
ACC	10,000	10,000	10,000
DEC	10,000	10,000	10,000
JK	0	0	0
INSTOP_DEC	100,000	100,000	100,000
Alarm			
ALM_EN	ALM_DIS	ALM_DIS	ALM_DIS
ALM_LOGIC	ALM_ACT_L...	ALM_ACT_L...	ALM_ACT_L...
ALM_MODE	ALM_IMM...	ALM_IMM...	ALM_IMM...

- 查看轴的状态：轴状态

轴状态															
轴	描述	STATE	SVON	EL+	EL-	ORG	ALM	SEL+	SEL-	INP	EMG	DPOS	MPOS	DSPEED	RUN_ERROR
轴(0)	轴(0)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(1)	轴(1)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(2)	轴(2)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(3)	轴(3)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(4)	轴(4)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(5)	轴(5)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(6)	轴(6)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS
轴(7)	轴(7)	READY	■	■	■	■	■	■	■	■	■	0	0	0	SUCCESS

- 查看 DI/DO 状态：數位輸入輸出檢視器  
可查看指定 DI/DO 的状态，可点击切换 DO 的状态。

数字输入输出查看器

DIN	描述	DIN	描述
0	Axis-0 IN1/LTC of MAS-...	8	Axis-2 IN1/LTC of MAS-...
1	Axis-0 IN2/RDY of MAS-...	9	Axis-2 IN2/RDY of MAS-...
2	Axis-0 IN4/JOG+ of MA...	10	Axis-2 IN4/JOG+ of MA...
3	Axis-0 IN5/JOG- of MAS...	11	Axis-2 IN5/JOG- of MAS...
4	Axis-1 IN1/LTC of MAS-...	12	Axis-3 IN1/LTC of MAS-...
5	Axis-1 IN2/RDY of MAS-...	13	Axis-3 IN2/RDY of MAS-...
6	Axis-1 IN4/JOG+ of MA...	14	Axis-3 IN4/JOG+ of MA...
7	Axis-1 IN5/JOG- of MAS...	15	Axis-3 IN5/JOG- of MAS...

DOUT	描述	DOUT	描述
0	Axis-0 OUT4/CAMDO of...	8	Axis-2 OUT4/CAMDO of...
1	Axis-0 OUT5/CMP of M...	9	Axis-2 OUT5/CMP of M...
2	Axis-0 OUT6/SVON of ...	10	Axis-2 OUT6/SVON of ...
3	Axis-0 OUT7/ERC of MA...	11	Axis-2 OUT7/ERC of MA...
4	Axis-1 OUT4/CAMDO of...	12	Axis-3 OUT4/CAMDO of...
5	Axis-1 OUT5/CMP of M...	13	Axis-3 OUT5/CMP of M...
6	Axis-1 OUT6/SVON of ...	14	Axis-3 OUT6/SVON of ...
7	Axis-1 OUT7/ERC of MA...	15	Axis-3 OUT7/ERC of MA...

Axis-2

- 查看/設置 VR 值：VR 表

可查看/設置 VR 的值，並可為每個自訂變數設置初始值。若使用在 Modbus 中，可設置對應 Modbus 的起始位置以及 Modbus 中的資料類型。

VR表					
名称	当前值	描述	初始值	Modbus	数据类型
范围[10-...		按鍵			
范围[60-...		監控區			
范围[20-...		板卡参数			
VR(200)	20	VL	20	40201	BIT_32_F...
VR(201)	100	VH	100	40203	BIT_32_F...
VR(202)	1000	ACC	1000	40205	BIT_32_F...
VR(203)	0		0	40207	BIT_32_F...
VR(204)	0		0	40209	BIT_32_F...
范围[30-...		用戶参数			
范围[40-...		坐标系设置			
范围[42-...		MARK设置			
范围[44-...		凸轮设置			
范围[46-...		准备位置			
VR(460)	179.86	指定位置X	179.86	40461	BIT_32_F...
VR(461)	139.35	指定位置Y	139.35	40463	BIT_32_F...
范围[48-...		其它位置信息			
VR(480)	0	X工作原点	0	40481	BIT_32_F...
VR(481)	0	Y工作原点	0	40483	BIT_32_F...
VR(482)	222	Z工作高度	222	40485	BIT_32_F...

- 查看變數值：觀察變數

選中變數後，可通過快速鍵 **Ctrl+W** 或者拖拉到觀察變數中添加變數。

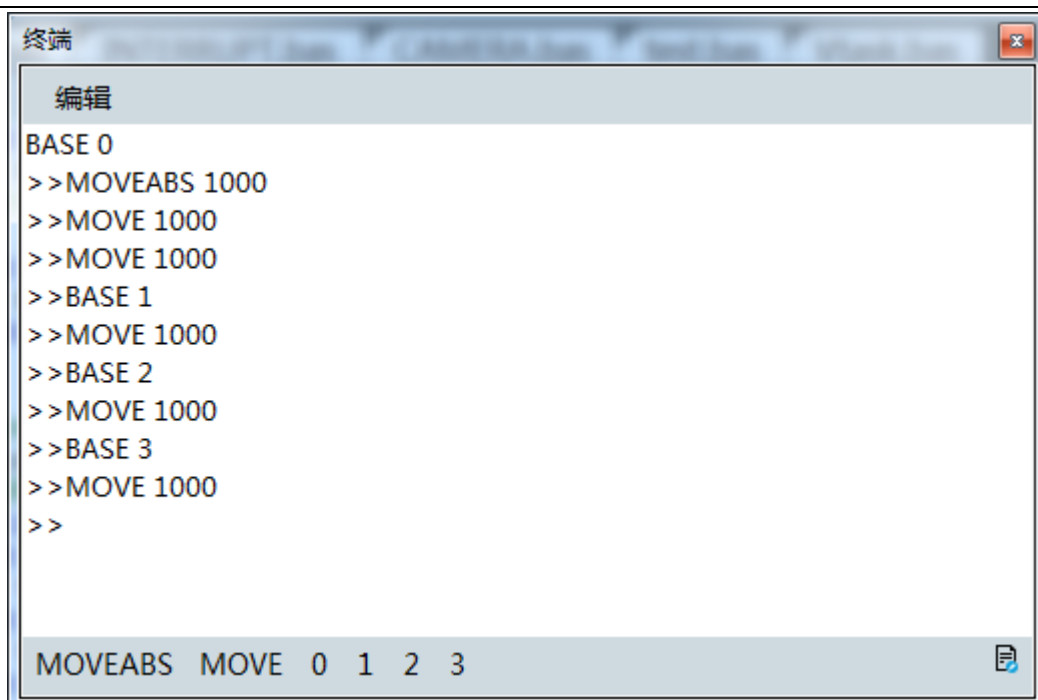
程式運行時，可（即時）查看變數（全域變數/區域變數/Motion 參數）值。

观察变量		
名称	值	上下文
VR(0)	0	VR
DPOS AX(0)	0	AXIS(0)
statusword	Nan	HMI.bas
STATE AX(0)	1	AXIS(0)

- 終端下指令：終端

可通過此工具下單行指令，進行相關測試/設定。

可直接輸入，也可事先設定巨集按鈕的命令，點擊巨集按鈕會直接下對應的命令。



## 3 主菜單

### 3.1 項目

提供專案操作相關功能，如新建工程（New），打開工程（Open）等，具體如下：

菜單名稱	操作說明
新建	創建一個新的工程並清除控制器的內容
導入	導入一個已存在的項目到控制器
更改	切換到另一個不同的項目，並與控制器的內容進行比較
從控制器中創建	根據控制器中的內容創建一個新的專案
保存	保存當前專案資訊
另存為	將當前專案另存為另一個工程
列印	列印當前活動編輯程式
預覽列印	預覽當前活動編輯程式
發佈	將已開發好的工程發佈（對 exe 加密，刪除原始程式碼等操作）
備份	打開 "備份管理 " 工具進行創建備份或管理備份的操作
關閉	關閉當前項目（此操作後將進入監控模式）
最近專案	列出最近操作的項目，可通過選擇相應項目進行項目切換
退出	退出應用程式

#### ● 發佈工程

工程已開發完畢，且已測試驗收。可通過此步驟發佈工程：將 exe 加密（綁定控制器硬體），將原始程式碼刪除等操作。

發佈後的工程，exe 第一次運行前需要先輸入密碼（正確輸入一次後，後續運行不需要輸入密碼）。密碼可通過 Modbus 通信輸入。

可通過 34017（5 位位址）/304017（6 位位址）的值判斷是否需要輸入密碼。

密碼對應的位址為 44601（5 位位址）/404601（6 位位址）起始的 20 個位址。

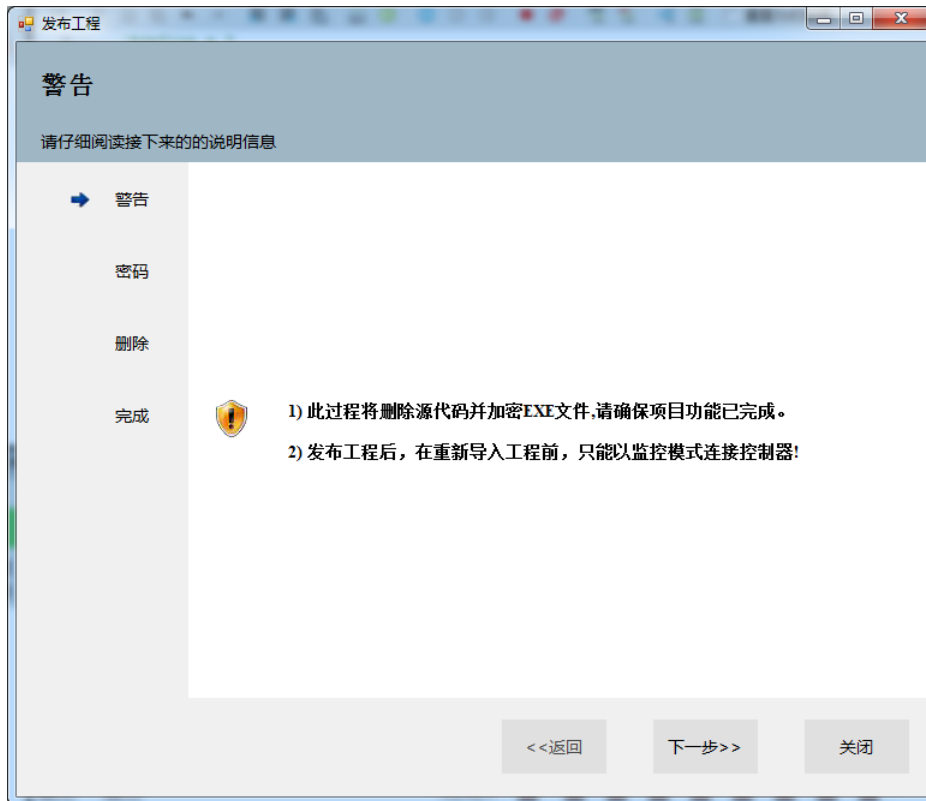
304017	Password Flag	ushort	0:No need to input password 1: Need to input password
--------	---------------	--------	--

MODBUS映射地址	名称	信号类型	信号地址	控制器所
404601		对应一个字符		
404602		对应一个字符		
...		对应一个字符		
404621		对应一个字符		

專案——>發佈，將跳出以下嚮導介面。

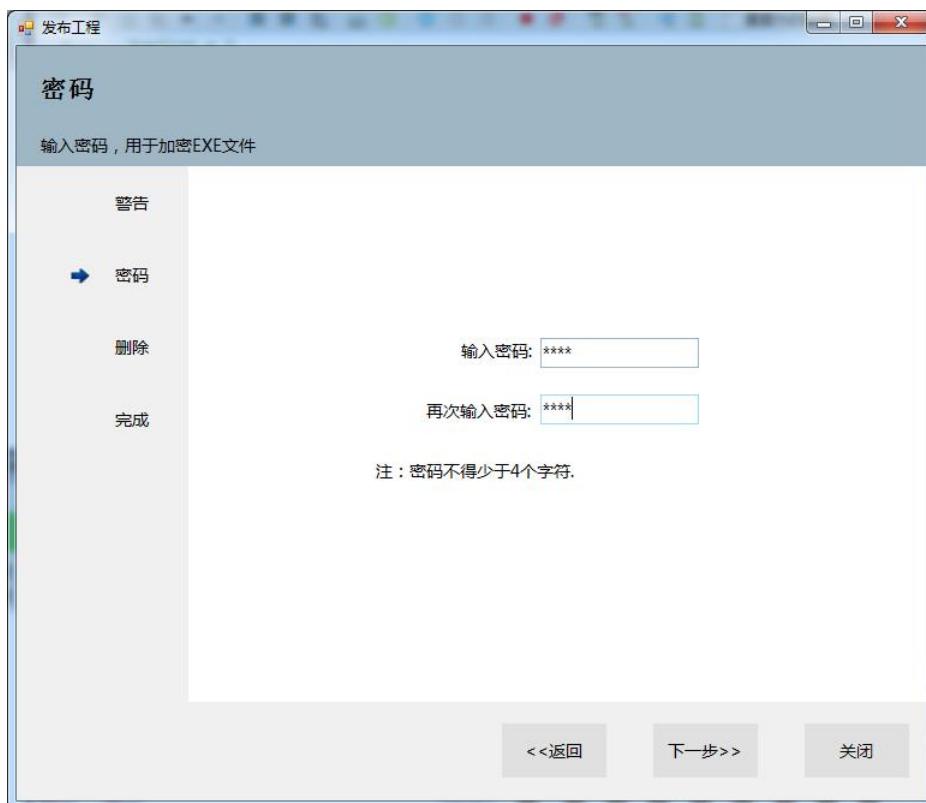
第一步：警告（提示）。

對此操作進行相關說明和警告。



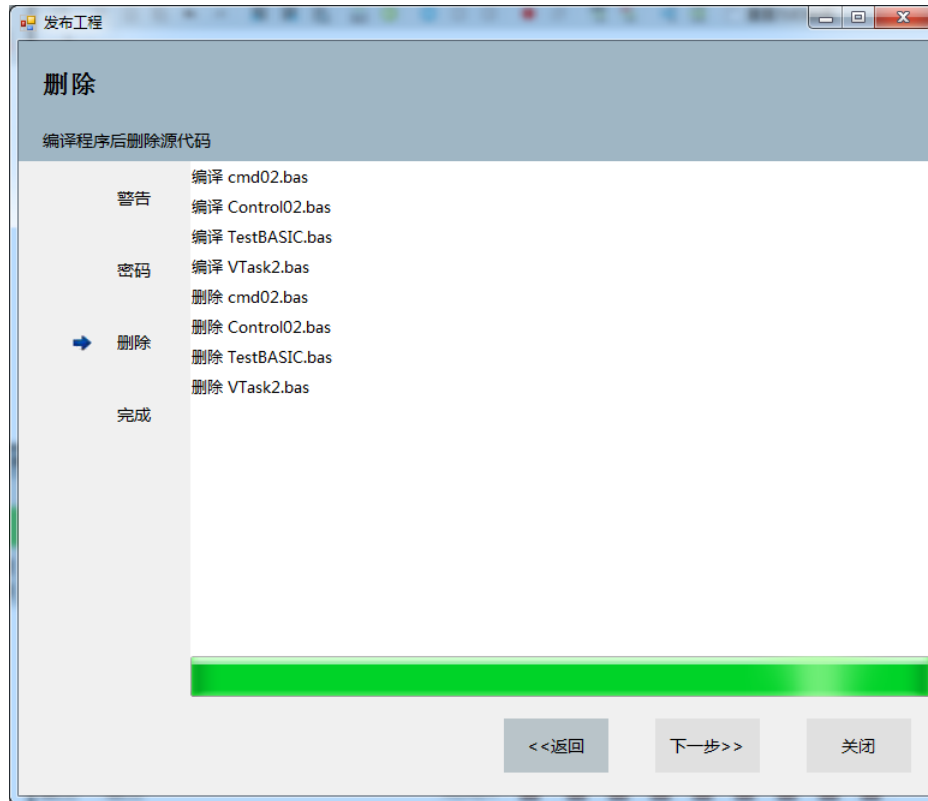
## 第二步：輸入密碼

此密碼將綁定控制器硬體，對編譯後的 exe 進行加密。

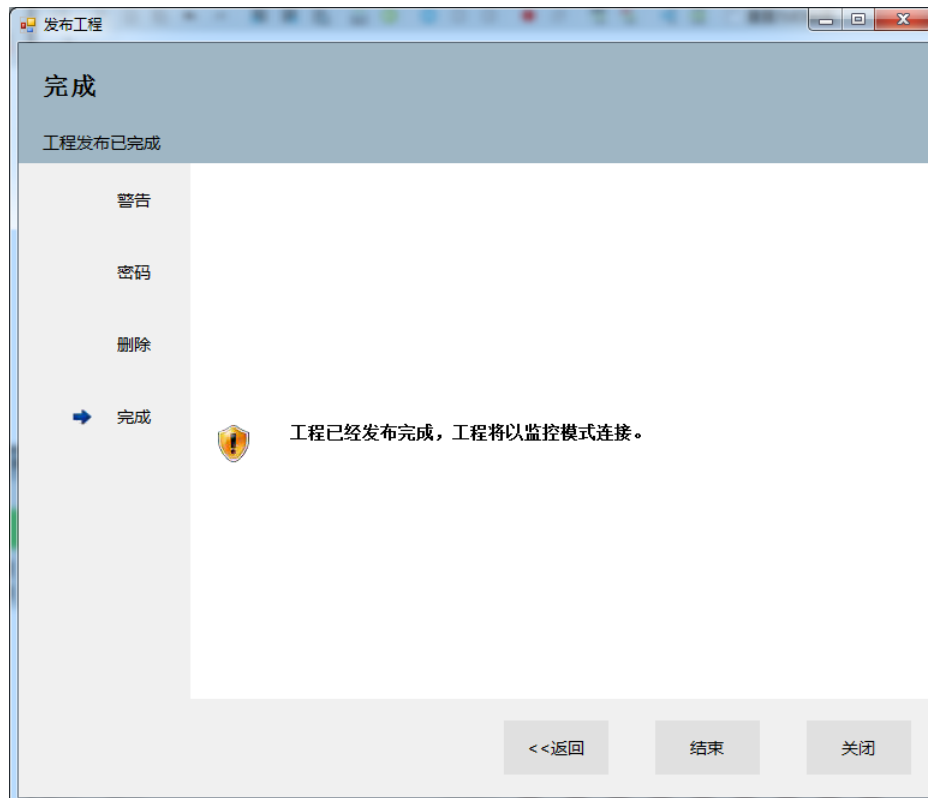


第三步：編譯後刪除原始程式碼

注：編譯失敗仍會刪除原始程式碼。因此，請確保在發佈工程前確認此工程所有工程已完成，且已驗收。



第四步：結束。提示將切換到監控模式。





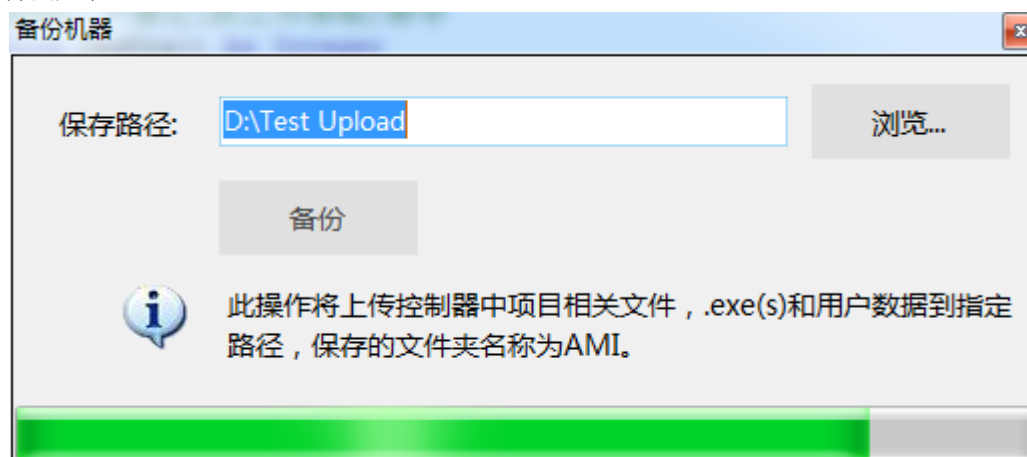
## 3.2 控制器

提供控制器操作相關功能，如操作模式變換，與控制器通信連接配置等，具體如下：

菜單名稱	操作說明
同步模式連接	與控制器進行同步模式連接
監控模式連接	與控制器進行監控模式連接
斷開連接	與控制器斷開連接
連接設置	更改與控制器通信的連接設置
重置控制器	重置控制器以進行暖開機
備份機器	將控制器中工程相關檔，.exe 和使用者資料等上傳到指定資料夾
目錄	清單顯示控制器上的程式相關資訊
日期時間	設置/查看控制器上日期和時間

### ● 備份機器

此操作會將控制器中的工程檔，exe 和使用者資料備份到指定路徑下，資料夾名稱為 AMI 資料夾。  
介面如下：




點擊【流覽】，確定保存的路徑後，點擊【備份】，將進行相關檔上傳。介面最下面為上傳的狀態（進度）。

### ● 目錄

清單顯示控制器上的程式相關資訊。

列名	含義
程式	程式的名稱
代碼類型	目前均為 BASIC
文件大小	此程式的大小，單位（位元組）
運行類型	即程式的類型：Normal（一般程式）或 Autorun（自動運行程式）
編輯	是否可編輯，目前均為可編輯圖示



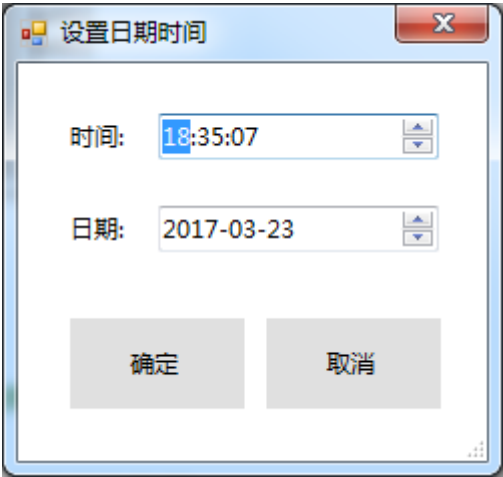
程序	代码类型	文件大小	运行类型	编辑
HMI.bas	BASIC	10,010	Normal	
RUN.bas	BASIC	6,689	Normal	
INTERRUPT.bas	BASIC	4,205	Normal	
CAMERA.bas	BASIC	2,164	Normal	
test.bas	BASIC	195	Normal	
Vtask.bas	BASIC	18	Normal	
DOControl.bas	BASIC	1,682	Normal	
ZControl.bas	BASIC	587	Normal	

- 日期和時間

設置/查看控制器上日期和時間



點擊“設置日期時間”，會彈出如下介面：



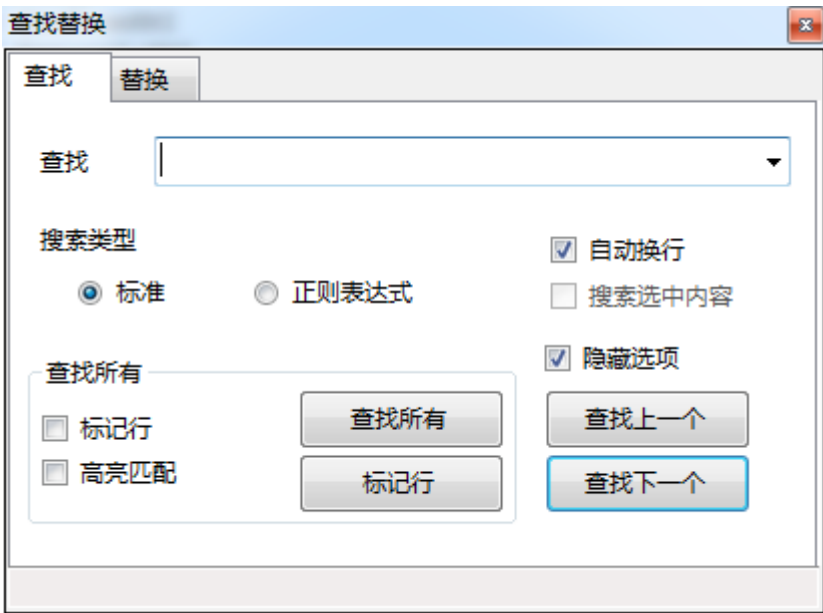
設置日期和時間後，點擊“確定”，將按照所設定值重新設置控制器的日期和時間。  
 點擊“與 PC 同步”，會設置控制器的日期和時間與 Studio 所在電腦一致。

### 3.3.編輯

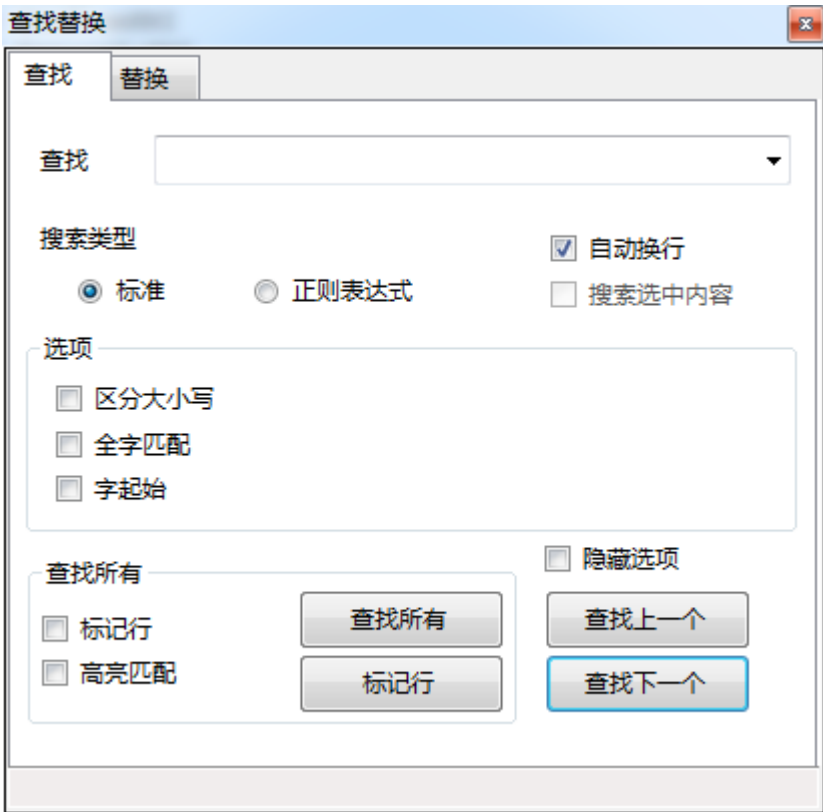
提供對編輯方塊操作相關功能，如剪切、複製、粘貼等，具體如下：

菜單名稱	操作說明
撤銷	撤銷最後一次的編輯操作
重做	重新操作最後一次取消的編輯操作
剪切	將當前選定的文本剪切到剪貼板
複製	將當前選定的文本複製到剪貼板中
粘貼	從剪貼板中粘貼文本
選擇所有	選擇文檔中的所有文本
選擇無	取消當前的選擇
查找和替換	使用“查找和替換”查詢/替換指定文字
轉到	跳轉到指定行
行號	顯示/取消顯示行號
書籤	書籤的操作包含以下： 切換書籤：在當前行添加/移除書籤 上一個書籤：跳轉到上一個書籤 下一個書籤：跳轉到下一個書籤 清除書籤：清除所有書籤
高級	高級操作包含以下： 注釋/取消注釋 增加/縮小行縮進 向前/向後導航 大寫/小寫選中內容

- 查找和替換



取消掉勾选“隐藏选项”，将显示完整的查找/替换表单：



### 3.4 程式

提供對程式相關操作功能，如添加新的/已存在的程式到當前專案，具體如下：

菜單名稱	操作說明
新建	新建一個空的程式

導入	導入一個已存在的程式到當前專案中
編輯	編輯當前專案中的一個/多個程式
調試	調試當前專案中的一個/多個程式
保存	保存當前程式
複製	複製當前專案中一個程式
重命名	重命名當前專案中的一個程式
刪除	刪除選中的當前專案中的程式
刪除所有	刪除當前專案中的所有程式
編譯所有	編譯當前專案中的所有程式，可選擇調試版或發佈版編譯
設置為自動運行程式	設置選中的當前專案中的程式為自動運行程式
設置為一般程式	設置選中的當前專案中的程式為一般程式
運行自動運行程式	運行所有設置為自動運行的程式
停止所有	停止所有正常運行的程式

## 3.5 構建/調試

提供對 BASIC 程式進行編譯、調試相關功能操作。具體如下：

菜單名稱	操作說明
編譯	編譯選中的程式（如有改動，會先保存）
運行（發佈版）	發佈版運行/停止當前活動的程式
運行（調試版）	調試版運行/停止當前活動的程式
單步（不進入副程式）	單步調試（不進入副程式）
繼續運行	繼續運行當前活動的程式
切換中斷點	在當前行上設置/移除中斷點
刪除中斷點	刪除所有中斷點
查看變數	將當前選中的變數加入到變數監測工具中

## 3.6 工具

提供相關工具以便於調試或查看控制器運行的狀態。具體如下：

菜單名稱	操作說明
軸參數	通過“軸參數”工具查看選定軸的屬性
軸運動測試	通過“軸測試”工具進行軸運動測試
軸狀態	通過“軸狀態”工具查看所有軸的狀態
數位輸入輸出表	通過“數位輸入輸出表”工具查看選定的數位輸入/輸出的狀態
氣油缸測試	通過“氣油缸測試”工具進行氣油缸相關測試
3D 軌跡	通過“3D 軌跡”工具可查看軸的 3D 運動軌跡
VR 表	通過“VR 表”工具查看/設定自訂變數

變數監測	通過“變數監測”工具查看變數值
功能列表	通過“功能列表”工具查看編輯器當前活動的 bas/ProjectDefine.bi 的 SUB/FUNCTION/TYPE
終端	通過“終端”工具可以與控制器進行 BASIC 指令交互
程式設計助手	通過“程式設計助手”可查看 BASIC 指令的簡單說明、用法和範例
自訂庫函數	通過“自訂庫函數”工具可以在 BASIC 中使用協力廠商 dll (C/C++) 的 API
選項	通過“選項”可對 Studio 的一些操作進行設置

具體請參看 9 工具的各小節。

## 3.7 視窗

顯示或隱藏相關介面/表單，具體如下：

菜單名稱	操作說明
工具列	顯示/隱藏工具列
狀態列	顯示/隱藏狀態列
控制器	顯示/隱藏控制器表單
項目	顯示/隱藏專案表單
輸出視窗	顯示/隱藏輸出視窗
重置表單佈局	重置為預設的表單佈局

## 3.8 幫助

提供說明文檔連結相關功能，具體如下：






菜單名稱	操作說明
Motion Studio 使用手冊	顯示 Motion Studio 使用手冊（注：需已安裝 PDF 閱讀器）
Motion BASIC 使用手冊	顯示 Motion BASIC 使用手冊（注：需已安裝 PDF 閱讀器）
MSConnection 使用手冊	顯示 C# Modbus 操作介面（MSConnection.dll）的使用手冊（注：需已安裝 PDF 閱讀器）
範本框架使用手冊	顯示範本框架的使用手冊（注：需已安裝 PDF 閱讀器）
關於 Motion Studio	對 Motion Studio 的版本/版權進行簡要說明

## 4 工具按鈕



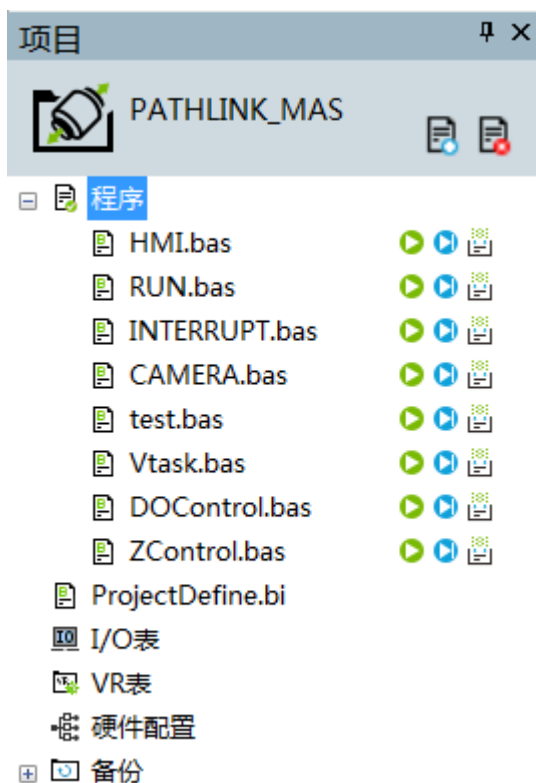
工具按鈕提供快捷使用常用功能/工具的方式。

圖示	功能	說明
	新建工程	創建一個新的工程並清除控制器的內容
	導入工程	導入一個已存在的項目到控制器
	保存工程	保存當前工程的所有資訊
	發佈工程	將已開發好的工程發佈（對 exe 加密，刪除原始程式碼等操作）
	同步模式連接	與控制器進行同步模式連接
	監控模式連接	與控制器進行監控模式連接
	斷開連接	與控制器斷開連接
	連接設置	更改與控制器通信的連接設置
	軸參數	顯示“軸參數”工具
	軸運動測試	顯示“軸測試”工具
	軸狀態	顯示“軸狀態”工具
	數位輸入輸出表	顯示“數位輸入輸出表”工具
	氣油缸測試	顯示“氣油缸測試”工具
	3D 軌跡	顯示“3D 軌跡”工具
	凸輪編輯器	顯示“凸輪編輯器”工具
	VR 表	顯示“VR 表”工具
	觀察變數	顯示“觀察變數”工具
	功能列表	顯示“功能清單”工具

	終端	顯示“終端”工具
	程式設計助手	顯示“程式設計助手”
	自訂庫函數	打開“自訂庫函數”工具
	選項	打開 “選項”對話方塊
	停止運動	停止所有 Task 運行以及運動



## 5 項目

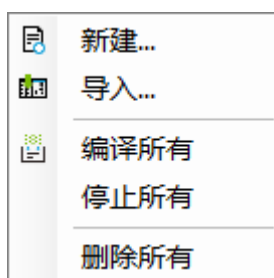


### 5.1 程式

程式節點下列出目前專案所包含的所有程式。

按一下每個程式名，同步模式下，編輯器將切換到對應的程式。可點擊圖示對相應程式進行編譯/發佈版運行/調試版運行的操作。

按一下程式，右鍵，會出現如下菜單：



可在專案中新建/導入程式。或者對當前已存在的程式進行編譯/停止或刪除的操作。

選中某個具體程式，右鍵，會出現如下菜單：



可對選中的項目進行編譯/運行/單步/繼續運行以及重命名/複製/刪除/設置為自動運行程式（設置為一般程式）等編輯操作。

## 5.2 ProjectDefine.bi

點擊此節點，會在編輯方塊中顯示 ProjectDefine.bi。

```

13 'CAMERA.bas: 用于处理与camera通讯
14 '常量宏定义
15 #DEFINE PI 3.14159265358979323846
16 '-----
17 '宏变量
18 #DEFINE IDLE 0
19
20 #DEFINE AXIS_X 0
21 #DEFINE AXIS_Y 1
22 #DEFINE AXIS_Z 2
23 #DEFINE AXIS_C 3
24 #DEFINE JOG_N 0
25 #DEFINE JOG_P 1
26 '操作步骤结束
27 #DEFINE STEPEND 200
28 '-----
29 '全局变量
30 #DEFINE controlword 1000
31 #DEFINE statusword 1001
32 #DEFINE preControlword 1002
33 #DEFINE preStatusword 1003
34 #DEFINE homeStep 1004
35 #DEFINE jogStep 1005
36 #DEFINE stopStep 1006
37 #DEFINE errorResetStep 1007
38 #DEFINE autoRunStep 1008
39 #DEFINE preAutoRunStep 1009
40 #DEFINE paraSetStep 1010
41 #DEFINE saveAllStep 1011
42
43 #DEFINE cameraRunStep 1012
44 '#DEFINE cameraOpenStep 1013
  
```

添加 bi 編輯器的功能有以下幾種：

1. 進行巨集定義

如在 ProjectDefine.bi 中添加以下巨集定義：

```
#DEFINE X_Pos 0
#DEFINE Y_Pos 1
#DEFINE Z_Pos 2
在.bas 中使用：
VR(X_Pos)=100
VR(Y_Pos)=200
VR(Z_Pos)=300
```

2. 自訂資料類型，如枚舉類型/結構體...

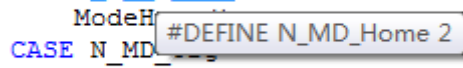
如在 ProjectDefine.bi 添加如下定義：

```
Type clr
    red As UByte
    green As UByte
    blue As UByte
End Type
在.bas 中使用：
Dim c As clr
c.red = 255
c.green = 128
c.blue = 64
```

**注：上述定義或聲明可使用在所有 Task 中。**

自 V1.5 版起，ProjectDefine 的巨集定義，在.bas 或 ProjectDefine.bi 中選中/按兩下巨集定義（字串），將跳出巨集定義資訊框，方便使用者查看/調試。

```
CASE N_MD_Idle
    Init()
CASE N_MD_Home
    ModeH
CASE N_MD_
    ModeORG()
CASE N_MD_Run
    ModeRun()
```



另，巨集定義字串加入自動補齊框，支援自動補齊功能。

## 5.3 I/O 表

創建硬體設定後，將自動生成對應的 I/O 表：

I/O表					
名称	当前值	描述	初始值	Modbus	数据类型
<b>Motion_MAS-324X Simulator_00</b>					
DOUT(0)	0	Axis-0 OUT...	0	1	BOOL
DOUT(1)	0	Axis-0 OUT...	0	2	BOOL
DOUT(2)	0	Axis-0 OUT...	0	3	BOOL
DOUT(3)	0	Axis-0 OUT...	0	4	BOOL
DOUT(4)	0	Axis-1 OUT...	0	5	BOOL
DOUT(5)	0	Axis-1 OUT...	0	6	BOOL
DOUT(6)	0	Axis-1 OUT...	0	7	BOOL
DOUT(7)	0	Axis-1 OUT...	0	8	BOOL
DOUT(8)	0	Axis-2 OUT...	0	9	BOOL
DOUT(9)	0	Axis-2 OUT...	0	10	BOOL
DOUT(10)	0	Axis-2 OUT...	0	11	BOOL
DOUT(11)	0	Axis-2 OUT...	0	12	BOOL
DOUT(12)	0	Axis-3 OUT...	0	13	BOOL
DOUT(13)	0	Axis-3 OUT...	0	14	BOOL
DOUT(14)	0	Axis-3 OUT...	0	15	BOOL
DOUT(15)	0	Axis-3 OUT...	0	16	BOOL
DIN(0)	0	Axis-0 IN1/L...	0	10001	BOOL
DIN(1)	0	Axis-0 IN2/...	0	10002	BOOL
DIN(2)	0	Axis-0 IN4/J...	0	10003	BOOL
DIN(3)	0	Axis-0 IN5/J...	0	10004	BOOL
DIN(4)	0	Axis-1 IN1/L...	0	10005	BOOL
DIN(5)	0	Axis-1 IN2/...	0	10006	BOOL
DIN(6)	0	Axis-1 IN4/J...	0	10007	BOOL
DIN(7)	0	Axis-1 IN5/J...	0	10008	BOOL
DIN(8)	0	Axis-2 IN1/L...	0	10009	BOOL

其中，列名與含義如下：

名稱：I/O 變數名，在程式中可直接使用

當前值：對應變數的當前值：0/1

描述：對應變數的描述

初始值：對應變數初始值，預設為 0。使用者可根據實際情況設置初始值，在程式運行前，可通過勾選“重置為初始值”將變數重新設置為初始值。

Modbus：對應變數的 Modbus 位址。

資料類型：目前只支援數位輸入/輸出，因此資料類型都是 BOOL 類型。

工具按鈕功能如下：



圖示	功能	說明
	定時創新	使能/禁用定時創新 I/O 表的當前值
	刷新	點擊後將刷新一次 I/O 表的當前值

	從.csv 檔中導入描述和初始值	從.csv 檔中導入已保存的 I/O 表的描述和初始值相關資訊顯示到 I/O 表中（ <b>注：描述欄中不能含逗號</b> ）
	描述和初始值保存為.csv 文件	將 I/O 表的描述和初始值相關資訊保存為.csv 文件
	初始值賦給當前值	點擊後將會把 DI(虛擬控制器才支援)/DO 變數的初始值設置為當前值
	下載到控制器	點擊後將會把 I/O 表下載到控制器，控制器端將會對 I/O 進行 Modbus 位址等匹配

## 5.4VR 表

VR 表即用戶自訂的全域變數表。

VR 變數的格式為 VR(x)，x 的值為 0-9999，即最大支持 10000 個。

VR 的資料類型默認為 Double 型，若 VR 是需要與 WOP 等進行關聯，則需要自行根據實際情況配置 VR 的資料類型與 Modbus 位址（注：此兩項必須與 WOP 上配置一致，否則可能出現 Modbus 通信不正常或者值顯示不正常的情況。）

VR 表如下：

VR表					
名称	当前值	描述	初始值	Modbus	数据类型
范围[10-...		按键			
范围[60-...		监控区			
范围[20-...		板卡参数			
VR(200)	20	VL	20	40201	BIT_32_F...
VR(201)	100	VH	100	40203	BIT_32_F...
VR(202)	1000	ACC	1000	40205	BIT_32_F...
VR(203)	0		0	40207	BIT_32_F...
VR(204)	0		0	40209	BIT_32_F...
范围[30-...		用户参数			
范围[40-...		坐标系设置			
范围[42-...		MARK设置			
范围[44-...		凸轮设置			
范围[46-...		准备位置			
VR(460)	179.86	指定位置X	179.86	40461	BIT_32_F...
VR(461)	139.35	指定位置Y	139.35	40463	BIT_32_F...
范围[48-...		其它位置信息			
VR(480)	0	X工作原点	0	40481	BIT_32_F...
VR(481)	0	Y工作原点	0	40483	BIT_32_F...
VR(482)	222	Z工作高度	222	40485	BIT_32_F...

其中，列名與含義如下：

名稱：變數名，在程式中可直接使用

當前值：對應變數的當前值

描述：對應變數的描述

初始值：對應變數初始值，預設為 0。使用者可根據實際情況設置初始值，在程式運行前，可通過勾選“重置為初始值”將變數重新設置為初始值。

Modbus：對應變數的 Modbus 位址。

資料類型：可在創建 Range 時，選擇資料類型或者單個配置（注：程式未運行時）

目前資料類型支持

BIT\_16\_UINT：16 位元無符號整型

BIT\_16\_INT：16 位元有符號整型

BIT\_32\_UINT：32 位元無符號整型

BIT\_32\_INT：32 位元有符號整型

BIT\_32\_FLOAT：32 位浮點數

- 工具按鈕



圖示	功能	說明
	新建範圍	點擊後將跳出子對話方塊，進行配置後創建新的 VR 範圍
	定時創新	使能/禁用定時刷新 VR 表的當前值
	刷新	點擊後將刷新一次 VR 表的當前值
	從.csv 文件中導入	從.csv 檔中導入已保存的 VR 相關資訊顯示到 VR 表中 (注：描述欄中不能含逗號)
	保存為.csv 文件	將 VR 表保存為.csv 文件
	初始值賦給當前值	點擊後會將所有 VR “初始值” 設置為當前值
	當前值賦給初始值	點擊後會將所有 VR 當前值設置為初始值
	下載到控制器	點擊後將會把 VR 表下載到控制器，控制器端將會對 VR 進行 Modbus 位址等匹配

新建範圍的對話方塊如下：

添加范围

从  到

Modbus数据类型:

Modbus起始地址:

VR 範圍標題列也可添加描述，如下圖：

名称	当前值	描述
范围[10-...		按键
范围[60-...		监控区
范围[20...		板卡参数
VR(200)	20	VL
VR(201)	100	VH
VR(202)	1000	ACC
VR(203)	0	
VR(204)	0	

## 5.5 硬體設定

點擊“配置”，可查看當前的硬體設定或者根據實際硬體情況配置專案所用板卡（運動控制卡+I/O 卡）。

配置分為三塊：輸入配置，輸出配置和運動配置。

● 輸入配置

輸入配置包含數位輸入配置和類比輸入配置（暫時保留）。目前主要支持

DAQ 卡：PCI-1750（16DI），PCI-1756（32DI），PCIE-1730（16DI）和 PCIE-1756（32DI）

Motion 卡：MVP-3245-MAS（16DI）和 PCI-1245LIO-MAS（16DI）

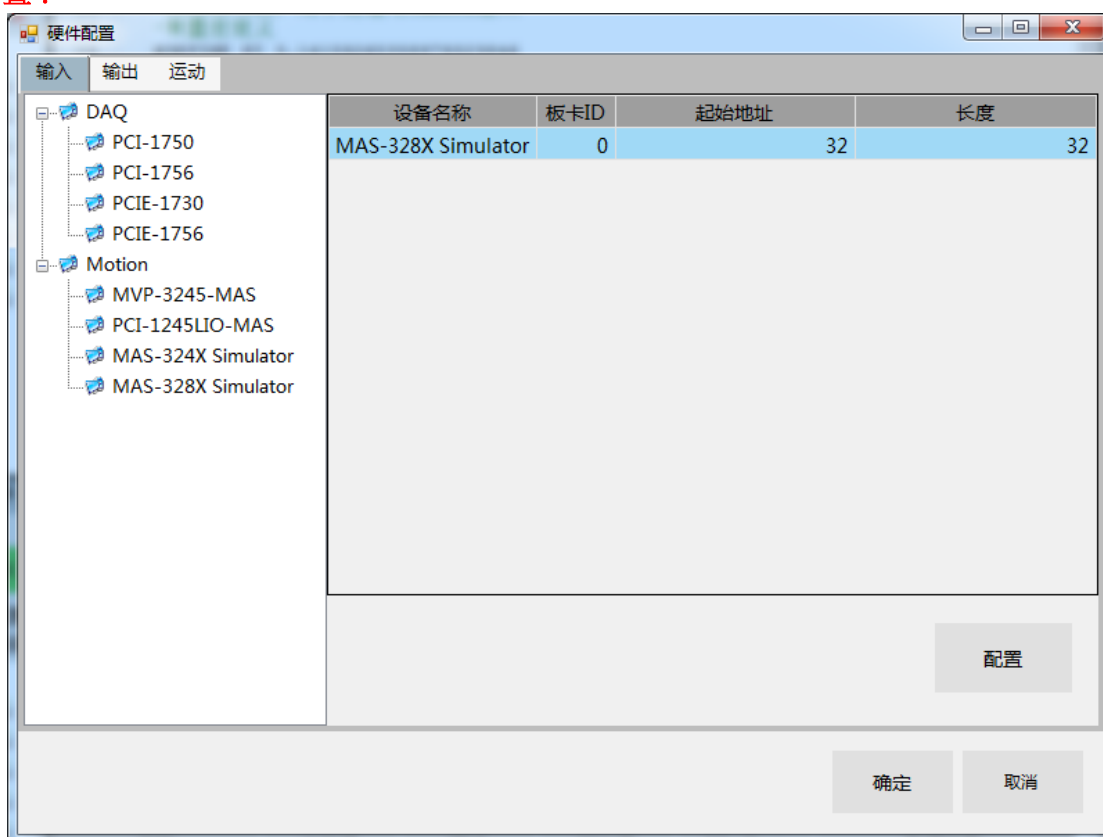
虛擬控制器：MAS-324X Simulator 和 MAS-328X Simulator

若自行配置，則選擇板卡類型後，點擊“添加”即可。

若是要查看配置，則按兩下配置行或者選中配置行，點擊“配置”即可。

若是要刪除配置，則選中配置行，點擊“刪除”。點擊“刪除所有”將刪除所有的輸入配置

**注：由於目前會根據實際可操作板卡創建設定檔，因此不建議使用者自行配置或刪除配置！**





### ● 輸出配置

輸出配置包含數位輸出配置和類比輸出配置（暫時保留）。目前主要支持 DAQ 卡：

DAQ 卡：PCI-1750(16DO)，PCI-1756(32DO)，PCIE-1730(16DO)和 PCIE-1756(32DO)

Motion 卡：MVP-3245-MAS（16DO）和 PCI-1245LIO-MAS（16DO）

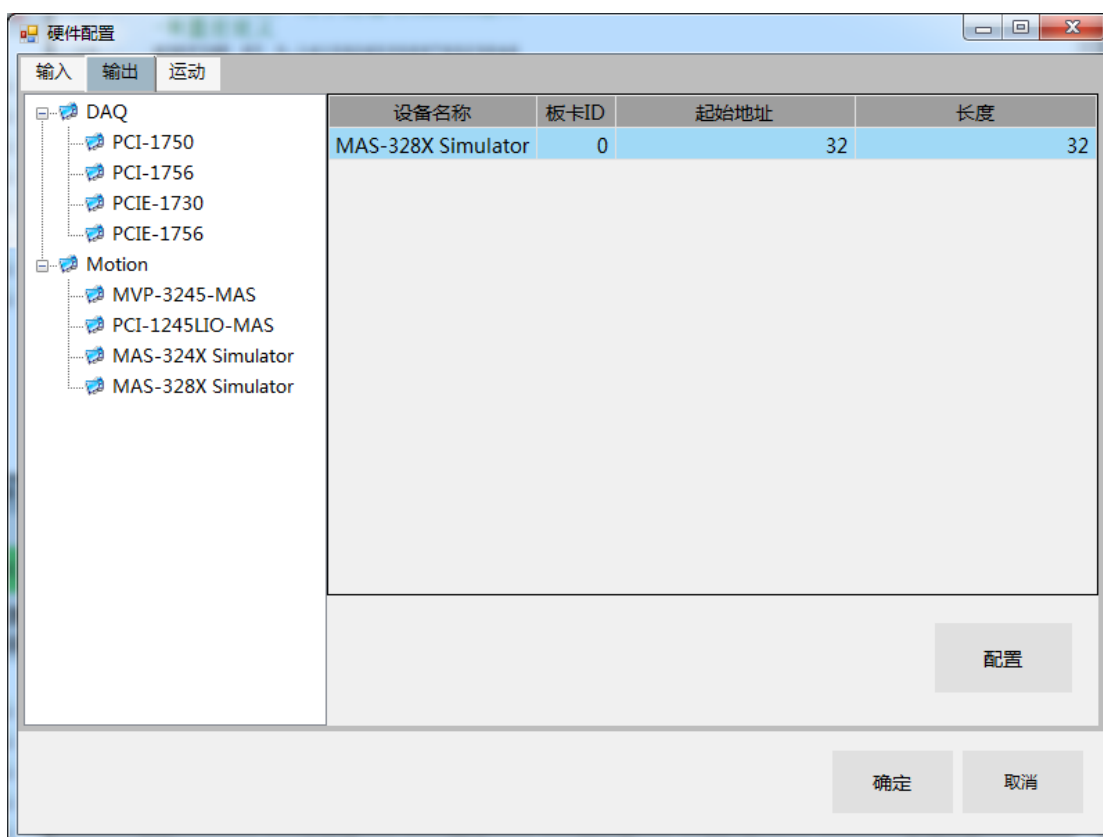
虛擬控制器：MAS-324X Simulator 和 MAS-328X Simulator

若自行配置，則選擇板卡類型後，點擊“添加”即可。

若是要查看配置，則按兩下配置行或者選中配置行，點擊“配置”即可。

若是要刪除配置，則選中配置行，點擊“刪除”。點擊“刪除所有”將刪除所有的輸出配置。

**注：由於目前會根據實際可操作板卡創建設定檔，因此不建議使用者自行配置或刪除配置！**



● 運動配置

運動配置主要是對運動控制卡的配置，其中自動包含對每個軸上的 DI/DO 的配置。目前主要支持：

實體板卡：PCI-1245-MAS、PCI-1285-MAS，PCI-1245L-MAS，PCI-1245LIO-MAS，MVP-3245-MAS 和 MVP-3285-MAS

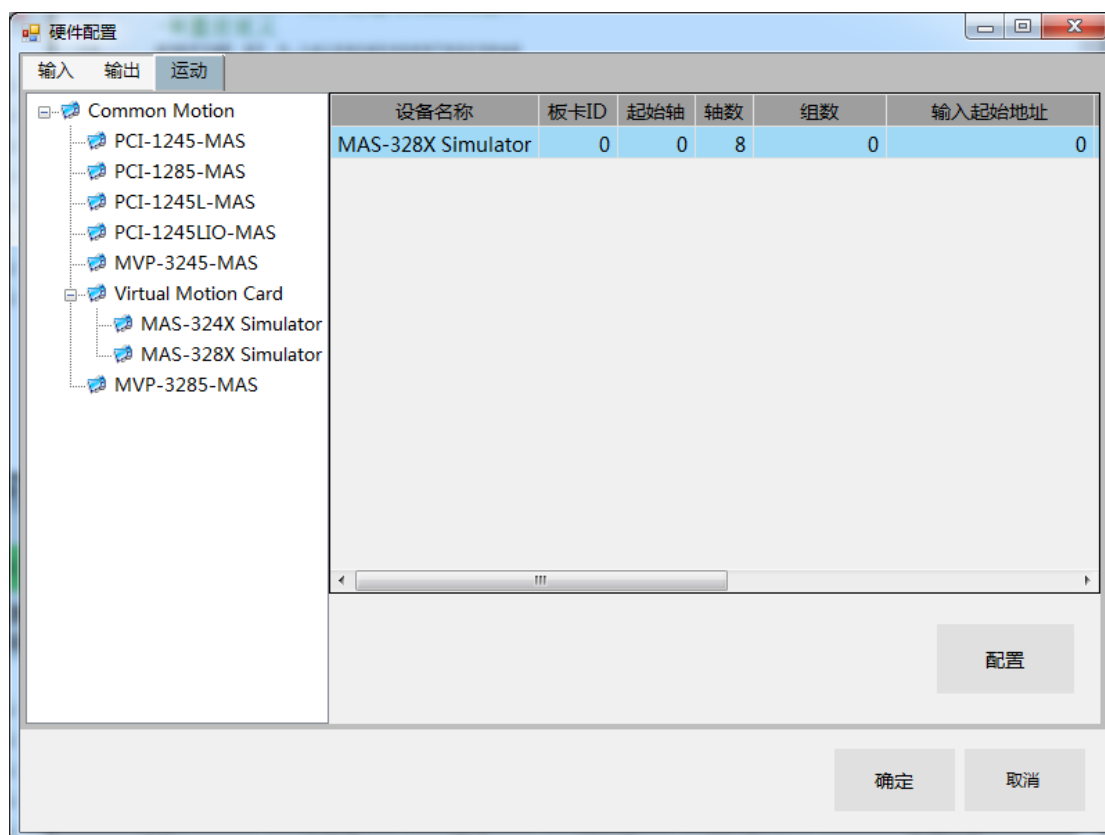
虛擬控制器：MAS-324X Simulator 和 MAS-328X Simulator

若自行配置，則選擇板卡類型後，點擊“添加”即可。

若是要查看配置，則按兩下配置行或者選中配置行，點擊“配置”即可。

若是要刪除配置，則選中配置行，點擊“刪除”。點擊“刪除所有”將刪除所有的運動配置。

**注：由於目前會根據實際可操作板卡創建設定檔，因此不建議使用者自行配置或刪除配置！**



點擊查看配置，可對運動控制卡是板卡配置或者軸的屬性進行配置，按兩下配置行，會跳出如下介面。

### 5.5.1 板卡的配置

MAS-328X Simulator的配置

Device

Axis

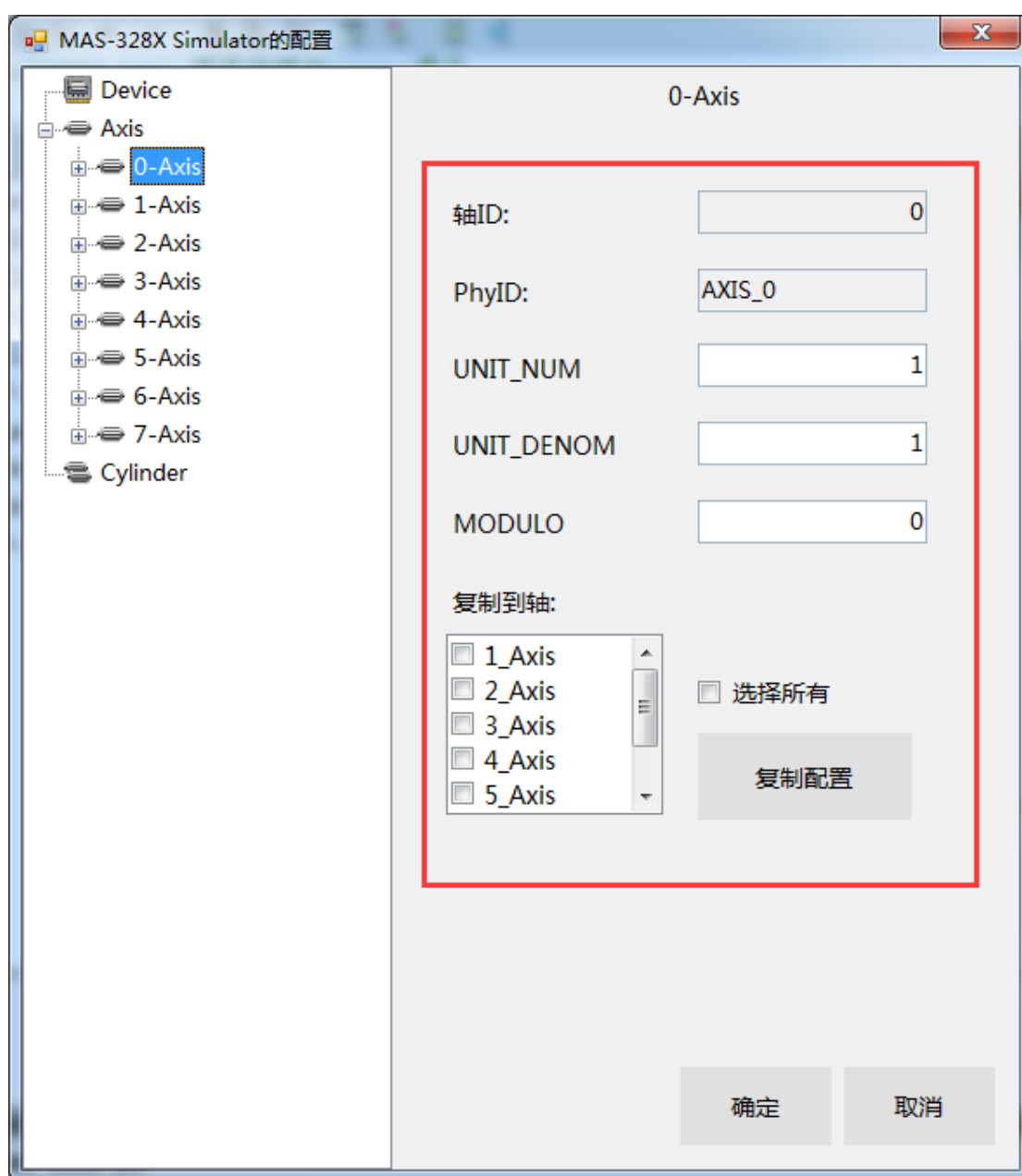
Cylinder

Device

板卡ID:	0
起始轴:	0
轴数:	8
最大组个数:	4
EMG逻辑电平:	EMG_ACT_LOW
EMG滤波时间:	EMG_5us
输入起始位置:	0
输出起始位置:	0
起始气油缸:	0
气油缸个数:	0
最大气油缸个数:	32

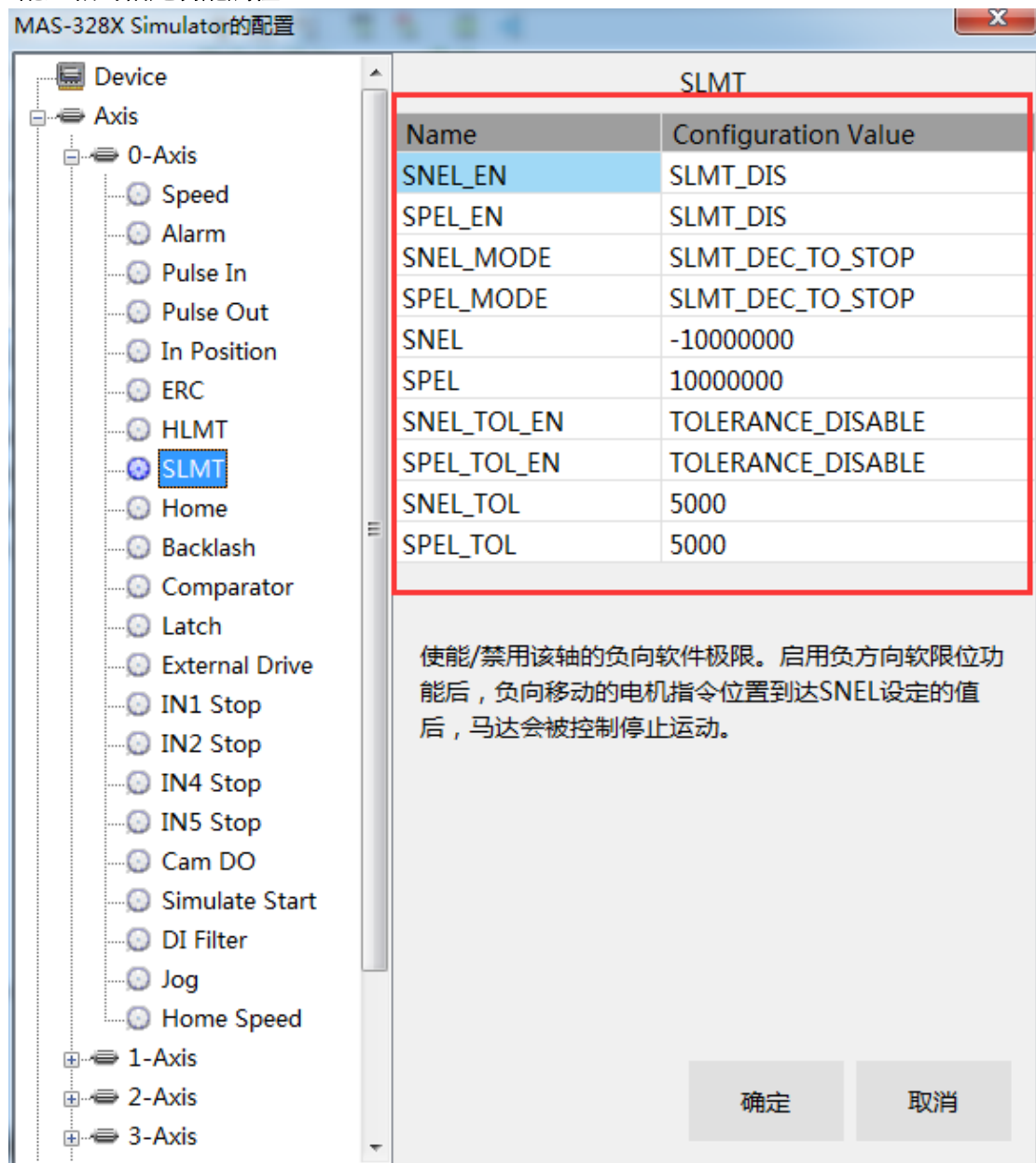
确定 取消

### 5.5.2 軸的基本配置



可選擇將此軸的配置複製到其他選中軸。

### 5.5.3 配置軸的指定功能屬性



### 5.5.4 配置氣油缸

- 新增

點擊氣油缸節點，點擊右鍵功能表【添加】，將跳出配置氣油缸個數的對話方塊，輸入個數，點擊【確定】，將按照預設配置產生對應個數的氣油缸子節點。



新增气油缸

新增个数:

已有个数:

最大个数:

确定 取消

之後可點擊氣油缸子節點對具體氣油缸進行相關配置。

MAS-324X Simulator配置

Device

- Axis
- Cylinder
  - 0-Cylinder
  - 1-Cylinder
  - 2-Cylinder
  - 3-Cylinder
  - 4-Cylinder
  - 5-Cylinder
  - 6-Cylinder
  - 7-Cylinder
  - 8-Cylinder
  - 9-Cylinder

0-Cylinder

气油缸ID:

气油缸别名:

气缸类型

☐ 单线圈电磁阀 输出0 输出1

☒ 双线圈电磁阀

前进 后退

到位方式:

输入:

延迟时间:  ms

报警时间:  ms

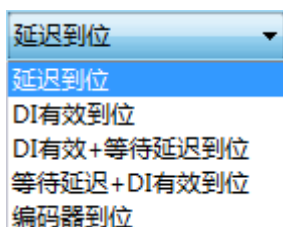
确定 取消

● 配置

氣油缸的配置頁面如下：

每個氣油缸可進行以下配置：

- 1) 氣油缸 ID：氣油缸序號。操作氣油缸時，調用氣油缸指令前通過 CYL\_BASE 傳入氣油缸序號指定要操作的氣油缸或者調用氣油缸指令時傳入氣油缸序號來指定。
- 2) 氣油缸別名：可根據功能配置氣油缸別名（底線、數位和字元組成），預設為 Cylinder。
- 3) 氣缸類型：單線圈電磁閥，對應一個 DO 控制；雙線圈電磁閥，對應 2 個 DO 控制
- 4) 控制 DO 的序號：根據選擇的氣缸類型，分別配置控制的 DO
- 5) 配置氣油缸的前進/後退的行為模式，共有以下 5 種到位方式



任意一種方式，都可配置報警時間：即指定時間未到位，則報警提示。

- a) 延遲到位：即延遲指定時間後到位，只需配置延遲時間

到位方式: 延迟到位

输入: DIN(0)

延迟时间: 5,000 ms

报警时间: 20,000 ms

- b) DI 有效到位：搭配指定 DI，DI 有效時氣缸到位

到位方式: DI有效到位

输入: DIN(0)

延迟时间: 5,000 ms

报警时间: 20,000 ms

- c) DI 有效+等待延遲到位：指定的 DI 有效後，延遲指定等待時間後氣油缸到位

到位方式: DI有效+等待延迟到位

输入: DIN(0)

延迟时间: 5,000 ms

报警时间: 20,000 ms

- d) 等待延遲+DI 有效到位：延遲指定等待時間，DI 有效時氣油缸到位

到位方式: 等待延迟+DI有效到位

输入: DIN(0)

延迟时间: 5,000 ms

报警时间: 20,000 ms

- e) 編碼器到位：關聯軸的編碼器，設定編碼器值，當指定的軸編碼器到位時氣油缸到位。

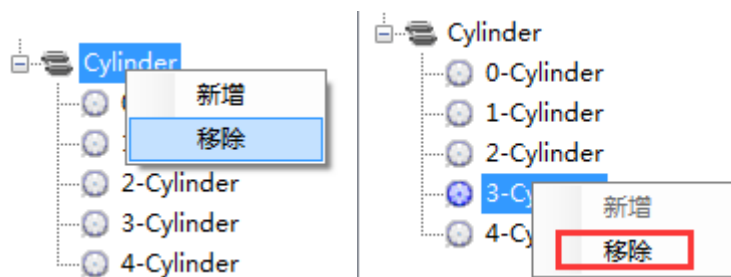


到位方式:	編碼器到位
轴:	0
編碼器值:	500 PPU
报警时间:	20,000 ms

注：氣油缸功能不能跨板卡，因此配置的 DI/DO/軸的序號只可在同一塊板卡對應的可用 DI/DO/軸的範圍中選擇。

#### ● 移除

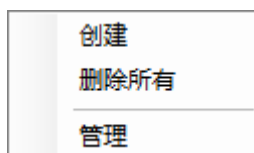
可點擊氣油缸節點，點擊右鍵菜單【移除】刪除所有氣油缸。也可選擇要刪除的氣油缸節點，點擊右鍵菜單【移除】，刪除對應的氣油缸及其配置。



添加或更新配置後，點擊“確定”將會保存配置，並將配置下載到控制器進行實體配置。

## 5.6 備份

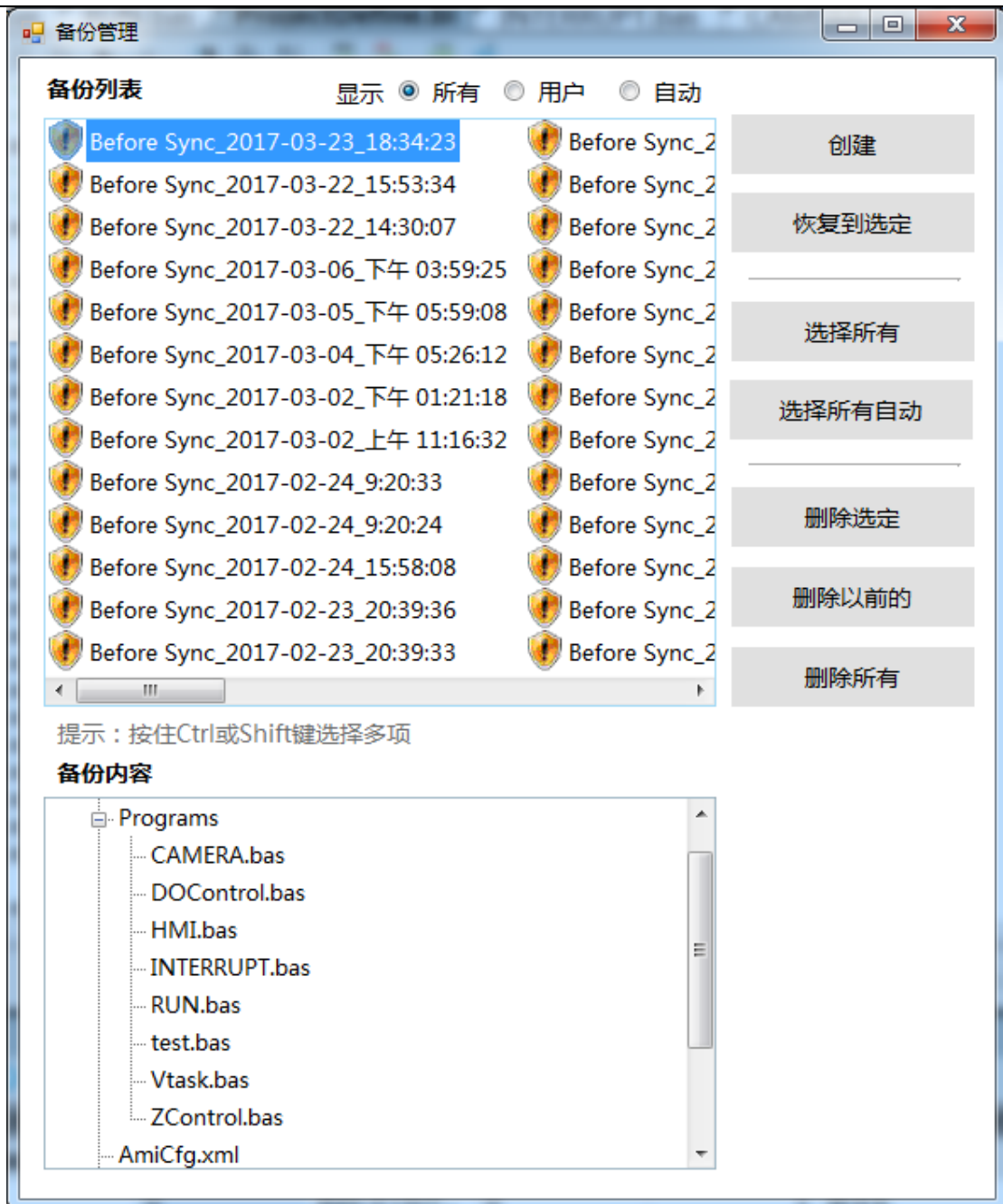
展開備份節點，將顯示所有備份資訊的子節點，點擊右鍵，會有以下右鍵菜單：



點擊“創建”將新建指定名稱的備份

點擊“刪除所有”將刪除所有備份

點擊“管理”，將跳出“備份管理”對話方塊，如下：



主體分為：備份清單，操作按鈕和備份內容三大部分。

其中：

1) 備份列表：根據上面的單選框顯示備份清單

所有：顯示所有備份清單

用戶：顯示所有使用者自訂的備份列表

自動：顯示所有 Studio 自動備份清單

2) 操作按鈕

創建：創建新的自訂備份

恢復到選定：選擇某一個備份，點擊後將恢復到選定備份

選擇所有：將選擇所有備份

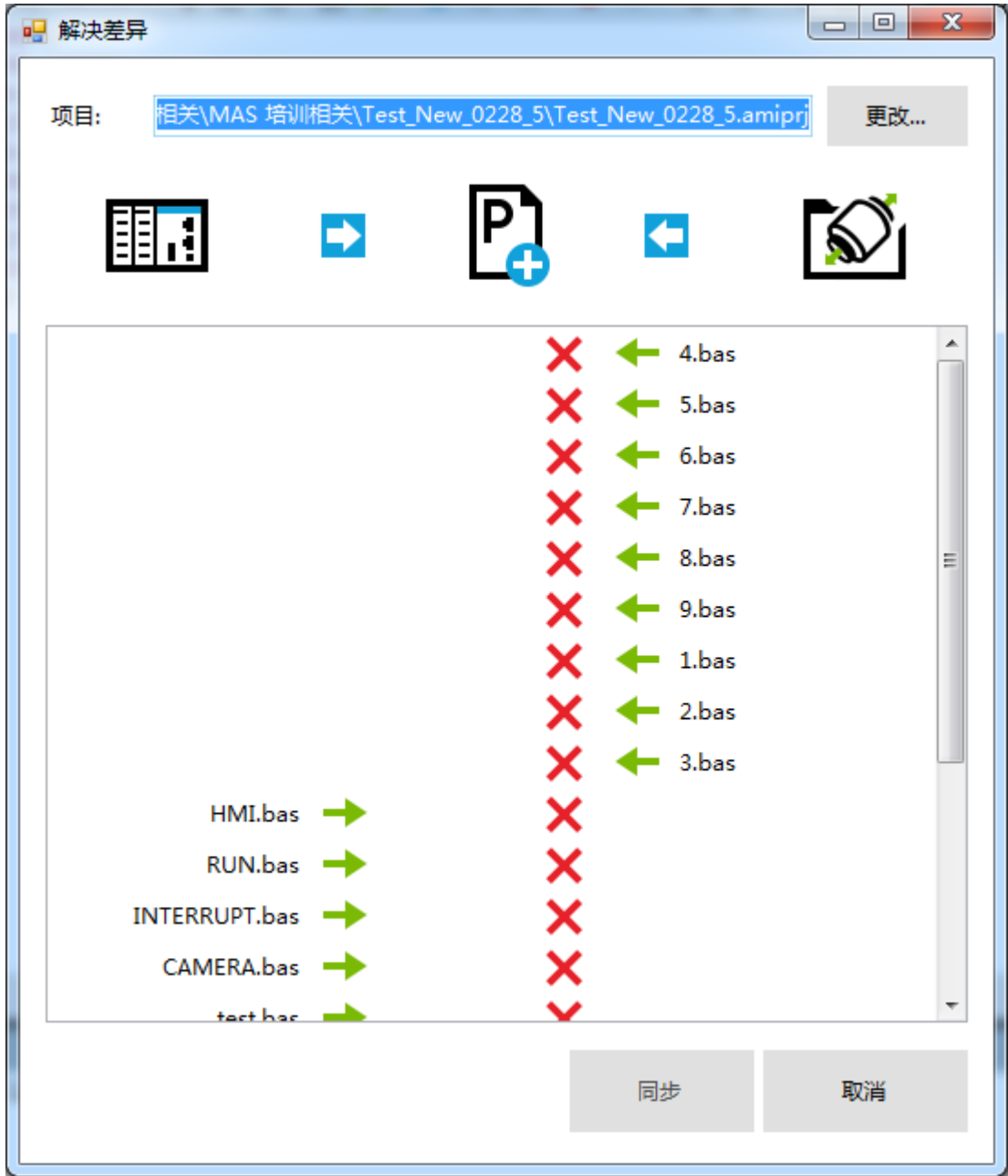
選擇所有自動：將選擇所有自動備份

刪除選定：選擇某一個備份，點擊後將刪除此備份

刪除以前的：選擇某一個備份後，點擊後將刪除此備份時間之前的所有備份  
刪除所有：將刪除所有備份






### 5.7 專案檢查

當更改專案時，切換到同步模式前，會先比較當前的專案與控制器中內容，如果兩者之間不相同，則會跳出以下對話方塊，解決差異。



項目：為目前選擇的專案，可通過點擊“更改”更改專案  
各個圖示及相關含義如下：

圖示	功能	說明
	新建工程	新建一個工程

	下載項目	控制器的內容與專案相同
	上傳項目	專案與控制器的內容相同
	下載檔案	下載檔案到控制器
	上傳文件	上傳檔到專案
	刪除檔	刪除檔
	比較文件	使用配置的比較工具比較檔後下載

對不同項操作後，對應的圖示會灰階。所有不同處都處理後，點擊“同步”進行項目端與控制器端的內容同步。

特別說明：

不同處比較除了使用者編輯的程式外，還有設定檔（對應配置模組，檔案名為 **AmiCfg.xml**），輸入輸出變數資訊（對應 I/O 表，檔案名為 **AmiVarDetail.xml**），自訂變數資訊（對應 VR 表，檔案名為 **AmiVR.xml**）以及關於自訂庫函數相應配置資訊。

由於輸入輸出變數資訊與設定檔是相互關聯的，因此列表中只列出 **AmiCfg.xml**；同理自訂庫函數相關資訊是相互關聯的，因此列表中只列出 **AmiCustomLib.bi**。也即：

**AmiCfg.xml**：配置資訊與輸入輸出變數資訊

**AmiVR.xml**：自訂變數資訊

**AmiCustomLib.bi**：自訂庫函數相關資訊

注：目前此三類檔不支持刪除和比較，只支持上傳/下載。

6 控制器



6.1 控制器相關操作

重置控制器：當控制器出現異常時，可通過點擊重置控制器。

停止運動：點擊後，所有運動將停止

使能伺服/禁用伺服：對伺服進行使能/禁用

重置錯誤：當運動軸卡的軸狀態為 Error\_Stopped 時，點擊後軸的狀態將恢復為 Ready

控制器狀態：

顏色	文字	含義
	狀態..OK	當下面的狀態節點的子節點 SYSTEM_ERROR 和 RUN_ERROR 的錯誤資訊都為 0,即沒有任何錯誤時)
	系統執行錯誤	當下面的狀態節點的子節點 SYSTEM_ERROR 和 RUN_ERROR 有錯誤時，但軸的狀態不是 DISABLED 或者 Error_Stopped 時
	運行狀態錯誤	軸的狀態為 DISABLED 或者 Error_Stopped 時

注意：由於上述的操作，實際是通過 BASIC 指令完成，因為可能會更改默認的 BASE 對應的軸（操作軸）。因為若是在終端下命令，需要再下 BASE 命令，指定操作軸。

## 6.2 節點

程式：同專案中的程式

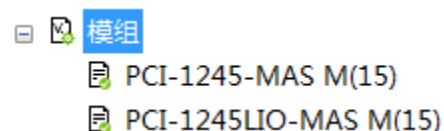
最大軸數：顯示目前可操作的最大軸數

狀態：顯示目前系統的錯誤資訊，目前包含系統錯誤（SYSTEM\_ERROR）和運行錯誤（RUN\_ERROR）兩個部分。其中 SYSTEM\_ERROR 為系統相關錯誤（對應指令 SYSTEM\_ERROR）；RUN\_ERROR 為運動相關的錯誤（包含所有軸的 RUN\_ERROR 指令資訊）

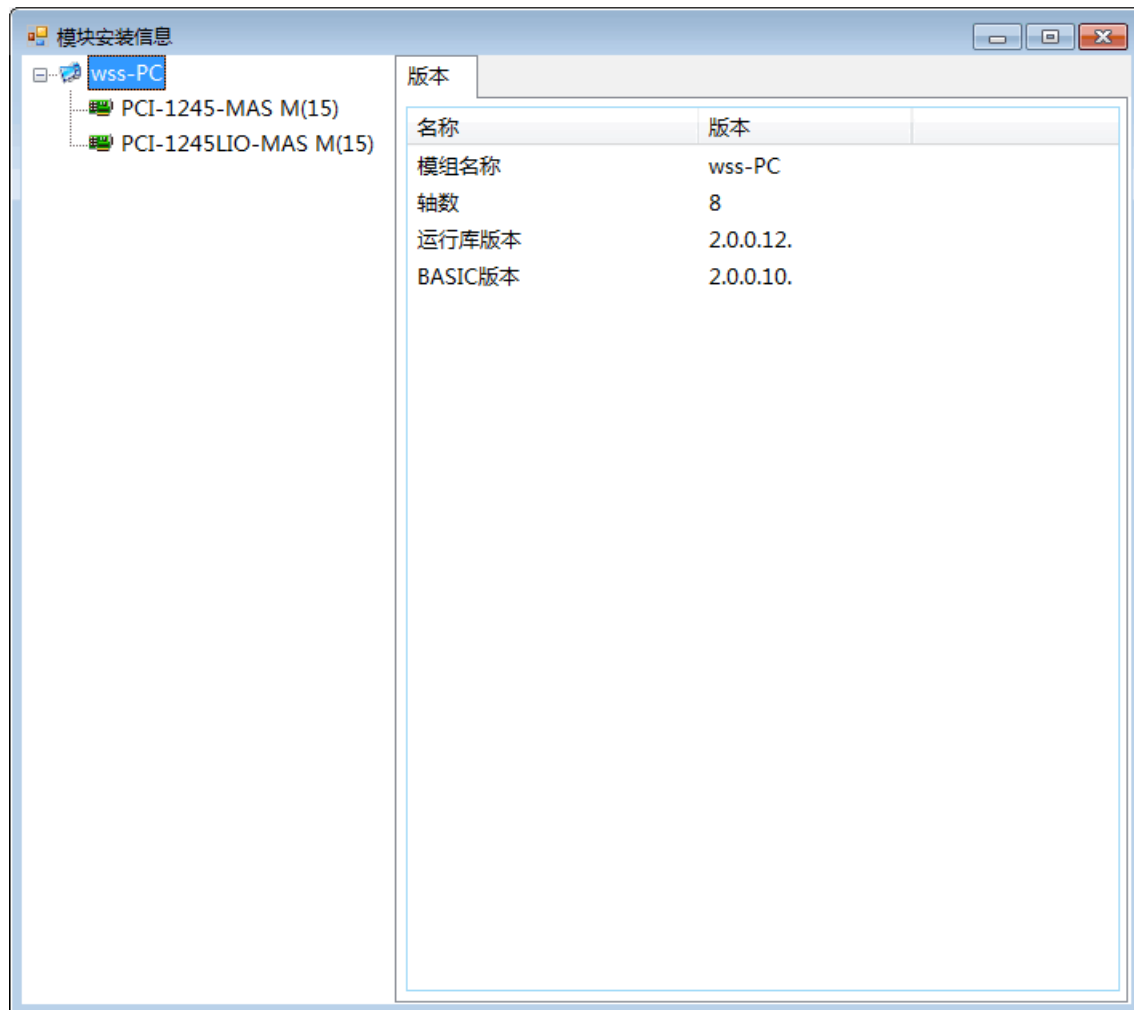
配置：



當只有一個板卡時，所有版本資訊將作為單個節點列出；當板卡個數超過兩個時，將會添加模組節點：



顯示所有可用的板卡，按兩下節點，會跳出對話方塊，顯示指定板卡的具體版本資訊：



## 7 程式編輯器

程式編輯器用於編輯 BASIC 程式檔，並可對 BASIC 程式編譯/調試/運行。

### 7.1 編輯功能

程式編輯器的編輯功能主要包含以下：

- 文字編輯（測試/重做/剪切/複製/粘貼/全選/不選）
- 關鍵字高亮顯示
- 智慧提示
- 高亮顯示當前行
- 顯示行號
- 查找/替換
- 跳轉到指定行
- 注釋/取消注釋選中行
- 行(塊)增加/減少縮進
- 向前/向後定位
- 行標識（書籤）
- 文本縮放：按住 Ctrl+滾輪上/下滾動

### 7.2 構建/調試功能

程式編輯器上的工具按鈕提供編輯/編譯/調試/運行的快捷方式。

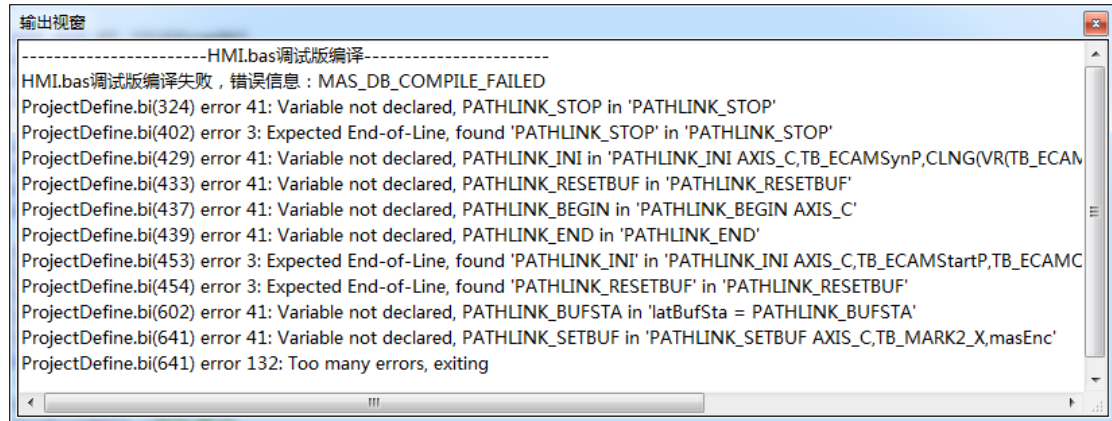


圖示	功能	說明
	保存	（若有改動）保存程式，並下載到控制器（同步模式）
	剪切	將當前選定的文本剪切到剪貼板
	複製	將當前選定的文本複製到剪貼板中
	粘貼	從剪貼板中粘貼文本
	撤銷	撤銷最後一次的編輯操作
	重做	重新操作最後一次取消的編輯操作
	向後導航	導航到上一次滑鼠位置
	向前導航	導航到下一次滑鼠位置



	查找	跳出“查找”對話方塊查找指定文字
	編譯	編譯當前活動的程式（如有改動，會先保存）
	發佈版編譯運行	發佈版運行/停止當前活動的程式
	調試版運行	調試版運行/停止當前活動的程式
	單步	單步調試（不進入副程式）
	繼續運行	繼續運行當前活動的程式
	切換中斷點	在當前行上設置/移除中斷點
	清除中斷點	刪除所有中斷點
	注釋	注釋當前選中內容
	取消注釋	取消注釋當前選中內容
	減少縮進	減少縮進當前選中行/塊
	增加縮進	增加縮進當前選中行/塊
	重置為初始值	此功能與 I/O 表，VR 表有關聯：當勾選此核取方塊時，在此程式運行前，會把設置 I/O 表和 VR 表中的變數值設置為對應的初始值（若未設置，則默認為 0）

## 8 輸出視窗



輸出視窗顯示所有提示資訊、錯誤資訊以及運行程式中所有的列印資訊（調試版運行時）。

## 9 工具

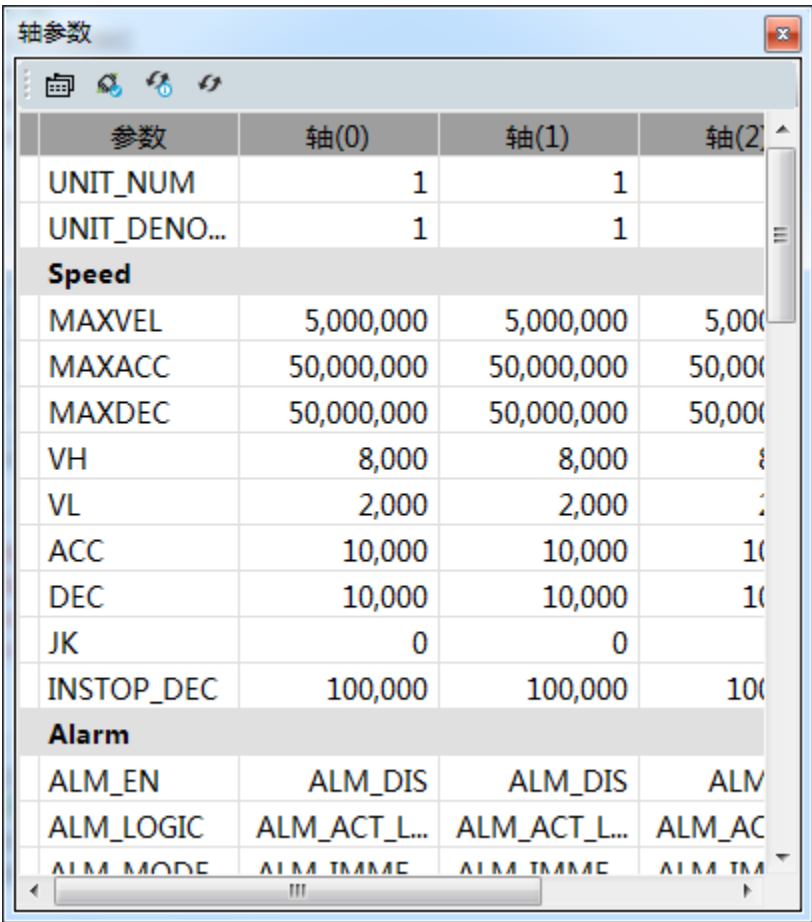
### 9.1 軸參數

軸參數工具用於查看選定軸的當前屬性值。

第一列：核取方塊若勾選，該屬性預設為正常顯示；若不勾選，則該屬性會隱藏顯示





第二列：參數名，與參數的 BASIC 關鍵字一致

後續每列為每個選定軸對應的屬性值。



参数	轴(0)	轴(1)	轴(2)
UNIT_NUM	1	1	
UNIT_DENO...	1	1	
<b>Speed</b>			
MAXVEL	5,000,000	5,000,000	5,000,000
MAXACC	50,000,000	50,000,000	50,000,000
MAXDEC	50,000,000	50,000,000	50,000,000
VH	8,000	8,000	8,000
VL	2,000	2,000	2,000
ACC	10,000	10,000	10,000
DEC	10,000	10,000	10,000
JK	0	0	0
INSTOP_DEC	100,000	100,000	100,000
<b>Alarm</b>			
ALM_EN	ALM_DIS	ALM_DIS	ALM_DIS
ALM_LOGIC	ALM_ACT_L...	ALM_ACT_L...	ALM_ACT_L...
ALM_MODE	ALM_TMME	ALM_TMME	ALM_TMME

工具按鈕

圖示	功能	說明
	顯示所有參數	切換是否顯示所有參數
	選擇軸	跳出對話方塊，顯示可操作軸序號，可更改選擇軸
	定時刷新	使能/禁用定時刷新
	刷新資料	點擊後刷新一次資料



9.2軸測試

軸測試工具用於測試選定軸運動是否正常。



按順序每列的列名與含義分別如下：

圖示	功能	說明
	軸	軸序號，軸名稱

	伺服	表示當前伺服的狀態，也可點擊切換 SVON/SVOFF
	負向點動運動	按下滑鼠開始運動，鬆開停止運動
	正向點動運動	按下滑鼠開始運動，鬆開停止運動
	選擇 PTP 運動 /Homing 運動	背景色為藍色的表示為當前的運動模式
	負向運動	根據前面的選擇進行負向 PTP 運動/Homing 運動
	正向運動	根據前面的選擇進行正向 PTP 運動/Homing 運動
	停止運動	停止該軸的當前運動（PTP 運動/Homing 運動）
	點到點距離	PTP 運動的參數：單次運動的距離
	DPOS	當前軸的理論位置
	配置運動參數	配置運動參數。可配置 PTP/Homing/Jog 運動的參數

點擊【選擇軸】，可顯示所有可操作軸，可切換操作軸（同軸參數工具中的選擇軸）。

## 9.3 軸狀態

軸狀態工具顯示所有可用軸的當前狀態（包含常用的 DI/DO 狀態）。

軸	描述	STATE	SVON	EL+	EL-	ORG	ALM	SEL+	SEL-	INP	EMG	DPOS	MPOS	DSPEED	RUN_ERROR
軸(0)	軸(0)	READY										0	0	0	SUCCESS
軸(1)	軸(1)	READY										0	0	0	SUCCESS
軸(2)	軸(2)	READY										0	0	0	SUCCESS
軸(3)	軸(3)	READY										0	0	0	SUCCESS
軸(4)	軸(4)	READY										0	0	0	SUCCESS
軸(5)	軸(5)	READY										0	0	0	SUCCESS
軸(6)	軸(6)	READY										0	0	0	SUCCESS
軸(7)	軸(7)	READY										0	0	0	SUCCESS

列名	說明
軸	軸序號，軸的名稱
描述	對軸的描述說明，默認為軸名稱
STATE	當前軸的狀態，具體含義請參數 BASIC 指令 STATE
SVON	當前軸的伺服狀態：ON/OFF，ON 為打開，OFF 為關閉
EL+	當前軸的正向硬體極限（EL+）的狀態
EL-	當前軸的負向硬體極限（EL+）的狀態
ORG	當前軸的原點（ORG）的狀態
ALM	當前軸的 Alarm 信號（ALM）的狀態

SEL+	當前軸的正向軟體極限（SEL+）的狀態
SEL-	當前軸的負向硬體極限（SEL-）的狀態
INP	當前軸的到位信號（INP）的狀態
EMG	當前軸的緊急停止信號（EMG）的狀態
DPOS	當前軸的理論位置
MPOS	當前軸的實際位置
DSPEED	當前軸的當前理論速度
RUN_ERROR	當前軸運動的最後錯誤，具體含義請參看 BASIC 指令 RUN_ERROR

當與虛擬控制器連接時，可通過點擊 EL+/EL-/ORG/ALM/INP/EMG 的圖示，對相應輸入信號置 1 或清零。

### 9.4數位輸入輸出表

數位輸入輸出表可查看指定 DI/DO 的狀態，可點擊切換 DO 的狀態。

数字输入输出查看器

DI/DO

DO

DIN	描述	DIN	描述
0	Axis-0 IN1/LTC of MAS...	8	Axis-2 IN1/LTC of MAS...
1	Axis-0 IN2/RDY of MAS...	9	Axis-2 IN2/RDY of MAS...
2	Axis-0 IN4/JOG+ of MA...	10	Axis-2 IN4/JOG+ of MA...
3	Axis-0 IN5/JOG- of MAS...	11	Axis-2 IN5/JOG- of MAS...
4	Axis-1 IN1/LTC of MAS...	12	Axis-3 IN1/LTC of MAS...
5	Axis-1 IN2/RDY of MAS...	13	Axis-3 IN2/RDY of MAS...
6	Axis-1 IN4/JOG+ of MA...	14	Axis-3 IN4/JOG+ of MA...
7	Axis-1 IN5/JOG- of MAS...	15	Axis-3 IN5/JOG- of MAS...

DOUT	描述	DOUT	描述
0	Axis-0 OUT4/CAMDO of...	8	Axis-2 OUT4/CAMDO of...
1	Axis-0 OUT5/CMP of M...	9	Axis-2 OUT5/CMP of M...
2	Axis-0 OUT6/SVON of ...	10	Axis-2 OUT6/SVON of ...
3	Axis-0 OUT7/ERC of MA...	11	Axis-2 OUT7/ERC of MA...
4	Axis-1 OUT4/CAMDO of...	12	Axis-3 OUT4/CAMDO of...
5	Axis-1 OUT5/CMP of M...	13	Axis-3 OUT5/CMP of M...
6	Axis-1 OUT6/SVON of ...	14	Axis-3 OUT6/SVON of ...
7	Axis-1 OUT7/ERC of MA...	15	Axis-3 OUT7/ERC of MA...



其中，每一塊 DI/DO 的顯示都以 Byte 為單位，即 8 個 DI/DO。

列名：DIN/DOUT 表示為數位輸入或者輸出，值即為此數位輸入/輸出序號。

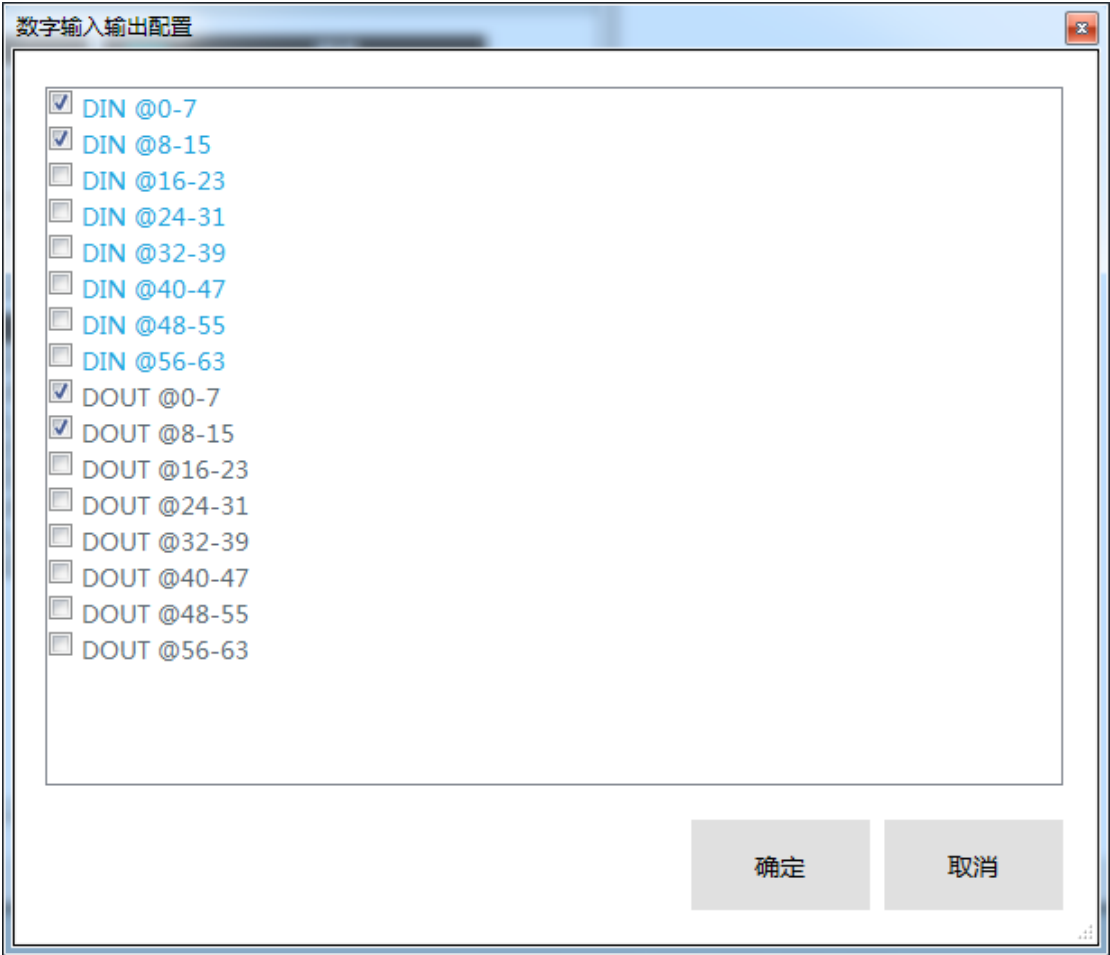
描述：對應數字輸入/輸出的說明。此說明與 I/O 表中的說明一致，可在 I/O 表中對應欄編輯，此處唯讀。

工具按鈕

圖示	功能	說明
----	----	----

	配置	配置要顯示的 DIN/DOUT
	顯示/隱藏	切換顯示/隱藏描述欄

點擊配置，彈出的對話方塊如下：



## 9.5VR 表

具體請參見 5.3 VR 表

## 9.6觀察變數

觀察變數工具用於查看變數（全域變數/區域變數/Motion 參數）值。



观察变量

名称	值	上下文
VR(0)	0	VR
DPOS AX(0)	0	AXIS(0)
statusword	Nan	HMI.bas
STATE AX(0)	1	AXIS(0)

添加變數的方法

- 1) 在編輯器中選中變數後，按快速鍵 Ctrl+W
- 2) 在編輯器中選中變數後，拖拉到觀察變數中
- 3) 直接在觀察變數中的名稱欄寫入變數名

● 工具按鈕

圖示	功能	說明
	定時刷新	使能/禁用定時刷新
	刷新資料	點擊後刷新一次變數值

● 右鍵菜單

刪除变量

☒ 十进制
   
☐ 十六进制
   
☐ 二进制

功能	說明
刪除變數	從觀察變數中移除此變數
十進位	變數的值以十進位顯示
十六進位	變數的值以十六進位顯示
二進位	變數的值以二進位顯示

特別說明：

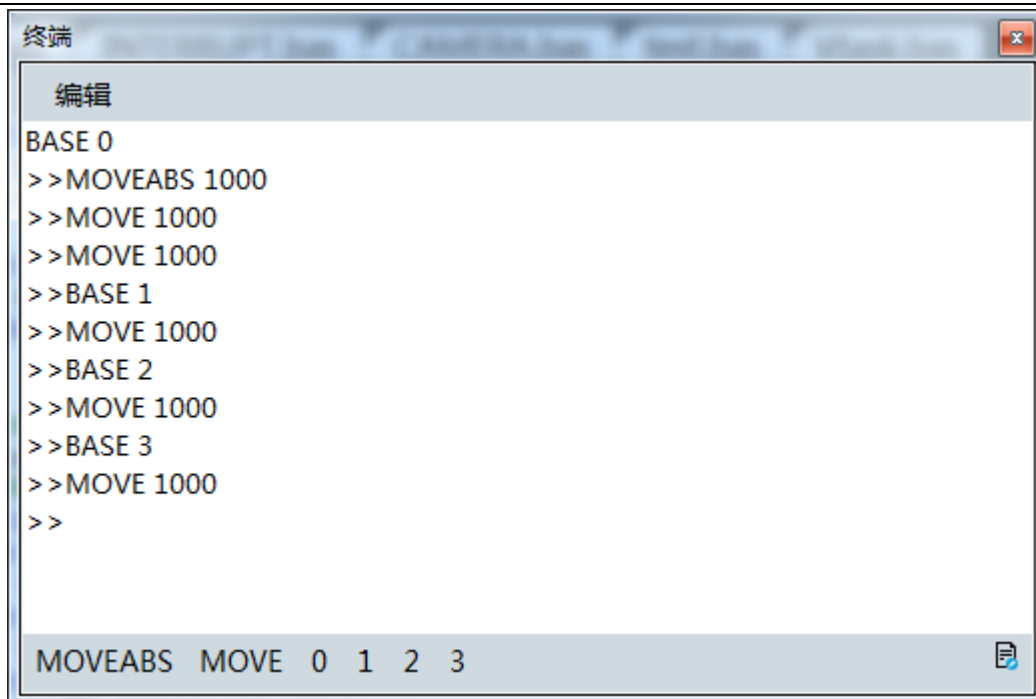
若查看軸的屬性/或狀態，需要在後面添加軸序號，比如查看 0 軸的 ACC 屬性，變數名為 ACC AX（0）。

### 9.7終端

終端提供使用者直接下 BASIC 指令的一種方式。

目前只支援單次一行指令。





指令可以為命令指令，如 **MOVE 1000**；也可以是問詢狀態或值的命令，如 **?ACC**，即為查詢操作軸的 **ACC** 屬性值。

注：若是對軸的操作或查看軸的屬性/狀態，需要先指定操作軸，即先下 **BASE** 指令，如 **BASE 0**。

終端的工具列（最下面）顯示的是巨集定義按鈕。點擊一次巨集按鈕，則終端下一次對應按到 **BASIC** 指令。點擊右下角的圖示，會出現如下對話方塊，可編輯巨集定義按鈕。



- 巨集按钮**  
 介面可分為三塊：  
 工具列預覽：即預覽編輯後在終端工具列（最下面）巨集定義按钮的顯示預覽，可通過拖拉巨集定義按钮更改順序  
 按钮清單：顯示已編輯的巨集定義按钮的標題與對應的 BASIC 語句

列名	說明
勾選	勾選用于表示此巨集按钮是否顯示在工具列
標題	巨集按钮的標題
命令	此巨集按钮的 BASIC 指令

注意：標題不能重複，否則會忽略。

操作按钮：

按钮	說明
添加	點擊後將添加一個巨集按钮
移除	選中一行後，點擊，將移除此巨集按钮
取消	取消此次編輯並關閉此表單

確定	確認此次編輯並關閉此表單
----	--------------

功能表項目

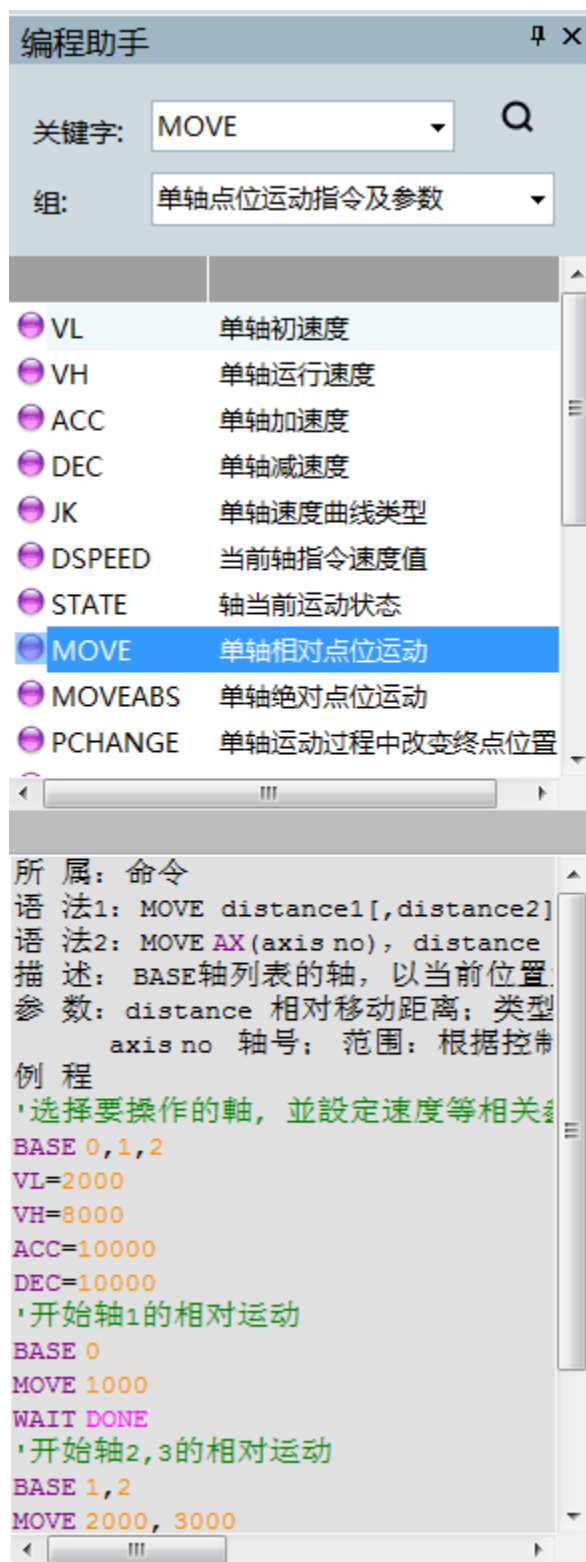
清除所有	Alt+C
复制	Ctrl+C
粘貼	Ctrl+V
复制，粘貼和运行	Ctrl+R

點擊編輯功能表或者在 Terminal 中右鍵，將跳出如下功能表項目

- 1) 清除所有，將清空目前 Terminal 的內容
- 2) 複製：複製目前選中的內容
- 3) 粘貼：將剪貼板的內容粘貼到 Terminal 中
- 4) 複製，粘貼和運行：將選中的內容複製，粘貼並運行

## 9.8程式設計助手

程式設計助手工具主要為方便使用者查詢 BASIC 指令的含義和使用，對每個指令有簡要的說明並提供列程幫助使用者理解和使用 BASIC 指令。



程式設計助手主介面可以分為三大塊：

- 1) 用戶可輸入關鍵字點擊查詢或者切換組，查看對應組別包含的所有關鍵字
- 2) 指令清單：對應組的所有指令以及每個指令的簡要說明
- 3) 指令說明與列程：輸入關鍵字，點擊查詢（若存在）或者直接在指令清單中選擇某個指令，下面將列出對應指令的語法，描述，參數以及常式，說明使用者理解和使用此 BASIC 指令。

關鍵字的輸入框可進行模糊匹配，即隨著輸入自動下拉列出模糊匹配到的關鍵字。

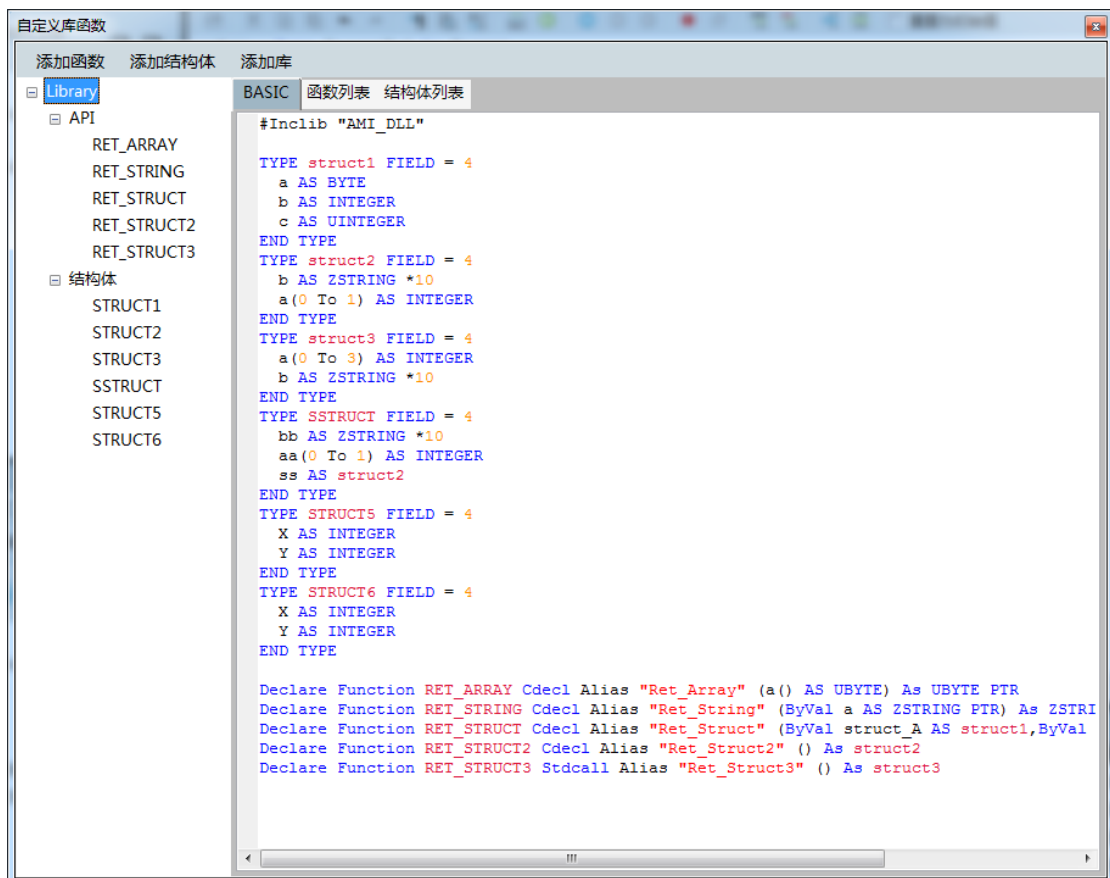
程式設計助手工具的顯示方式目前有兩種：

- 1) 菜單：工具——>程式設計助手
- 2) 在編輯方塊中選中關鍵字後，按 F12：將跳出編輯助手，且跳轉到對應關鍵字

## 9.9 自訂庫函數

為擴展支援其他相關硬體（如相機）的功能，Motion Studio 提供介面方便使用者自訂 C dll 中的 API 在 BASIC 中的關鍵字（即提供從 C 介面轉換為 BASIC 介面的功能）。

### ● 主介面



可分為以下三大塊：工具按鈕，自訂庫函數樹和內容表單

#### 1) 工具按鈕

添加函數：點擊後將跳出“添加函數”表單

添加結構體：點擊後將跳出“添加結構體”表單

添加庫：點擊後將跳出“添加庫函數”表單

#### 2) 自訂庫函數樹：列出目前已定義的自訂 API 和結構體

點擊“Library”節點，內容表單三個頁面：BASIC、函數清單和結構體清單

點擊“API”：內容表單將顯示“函數清單”表單

點擊“結構體”：內容表單將顯示“結構體清單”表單

點擊具體 API/結構體：內容表單將顯示此 API/結構體的具體配置資訊

#### 3) 內容表單

包含以下三個頁面

BASIC：自訂庫函數對應的 BASIC 介面

函數清單：顯示已定義的函數清單。按兩下對應函數行，可查看/編輯此函數

結構體清單：顯示已定義的 結構體清單。按兩下對應結構體行，可查看/編輯此結構體

● BASIC

BASIC 函数列表 结构体列表

```
#Inclib "AMI_DLL"

TYPE struct1 FIELD = 4
  a AS BYTE
  b AS INTEGER
  c AS UINTEGER
END TYPE
TYPE struct2 FIELD = 4
  b AS ZSTRING *10
  a(0 To 1) AS INTEGER
END TYPE
TYPE struct3 FIELD = 4
  a(0 To 3) AS INTEGER
  b AS ZSTRING *10
END TYPE
TYPE SSTRUCT FIELD = 4
  bb AS ZSTRING *10
  aa(0 To 1) AS INTEGER
  ss AS struct2
END TYPE
TYPE STRUCT5 FIELD = 4
  X AS INTEGER
  Y AS INTEGER
END TYPE
TYPE STRUCT6 FIELD = 4
  X AS INTEGER
  Y AS INTEGER
END TYPE

Declare Function RET_ARRAY Cdecl Alias "Ret_Array" (a() AS UBYTE) As UBYTE PTR
Declare Function RET_STRING Cdecl Alias "Ret_String" (ByVal a AS ZSTRING PTR) As ZSTR
Declare Function RET_STRUCT Cdecl Alias "Ret_Struct" (ByVal struct_A AS struct1,ByVal
Declare Function RET_STRUCT2 Cdecl Alias "Ret_Struct2" () As struct2
Declare Function RET_STRUCT3 Stdcall Alias "Ret_Struct3" () As struct3
```

● 函數清單

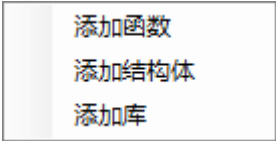
BASIC	函数列表	结构体列表		
BASIC名称	函数名称	函数库名	调用方式	参数个数
RET_ARRAY	Ret_Array	AMI_DLL.dll	C_call	2
RET_STRING	Ret_String	AMI_DLL.dll	C_call	2
RET_STRUCT	Ret_Struct	AMI_DLL.dll	C_call	3
RET_STRUCT2	Ret_Struct2	AMI_DLL.dll	C_call	1
RET_STRUCT3	Ret_Struct3	AMI_DLL.dll	Stdcall	1

● 結構體清單

BASIC	函数列表	结构体列表	
名称	对齐模数	成员个数	
struct1	pragma_pack_4	3	
struct2	pragma_pack_4	2	
struct3	pragma_pack_4	2	
SSTRUCT	pragma_pack_4	3	
STRUCT5	pragma_pack_4	2	
STRUCT6	pragma_pack_4	2	

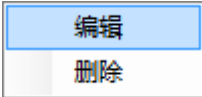
● 菜單

點擊 Library/API/結構體節點，點擊右鍵，會彈出以下右鍵菜單：



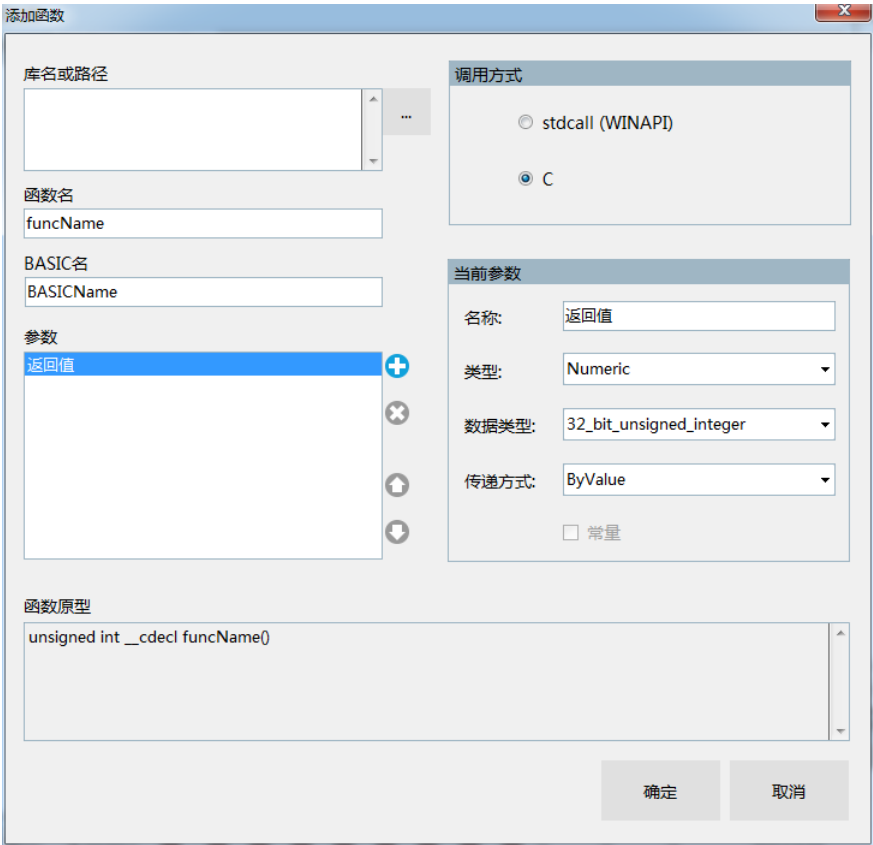
功能表功能與工具按鈕的功能一致。

點擊具體 API/結構體，點擊右鍵，會彈出以下右鍵菜單：



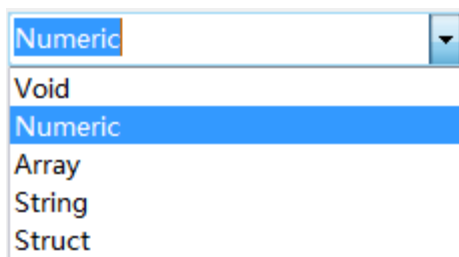
可對此 API/結構體進行編輯或者刪除。

● 添加函數



添加 API 提供以下輸入參數：

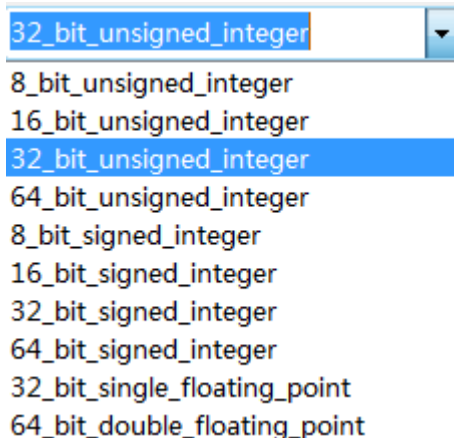
- 庫名或函數：輸入庫函數的名字（預設首先查找 Studio EXE 所在路徑；若 Studio EXE 沒有找到，則查找系統路徑）或者庫函數的所在路徑+庫函數的名字。
- 函數名：API 的名字（注：必須與 dll 中的 API 名字一致）。
- BASIC 名：BASIC 中定義的 API 名字（BASIC 中的關鍵字，注不能與已有的 BASIC 關鍵字重複）。
- 調用方式：選擇為 Stdcall 或者 C 調用方式，預設為 C 調用。
- 參數：API 中的參數，左邊列表列出已定義的參數名，選擇後，右邊為當前參數的具體資訊；可對參數進行添加/刪除/上移/下移的操作。
- 當前參數：顯示/編輯當前 Parameter 的資訊，包含名稱，類型，資料類型和傳遞方式。  
名稱：除第一個參數名默認為返回值（唯讀）外，其他函數名只能為字元，數位和底線。  
類型：此參數的類型，分為 Void（無，目前只支持返回值），Numeric（數位），Array（陣列），String（字串）和結構體（Struct）



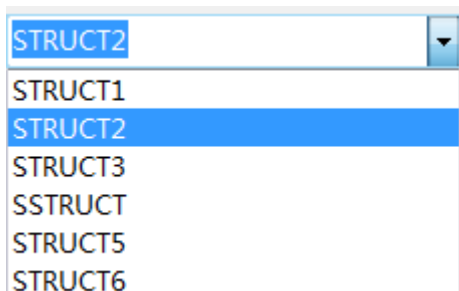
資料類型：根據所選類型而變化。

當類型為 Void 或者 String 時，沒有資料類型

當類型為 Numeric 或者 Array 時，資料類型有如下幾類：



當類型為 Struct 時，資料類型將列出已添加支援的結構體類型，如下：



常量：勾選表示參數為常量

- 函數：唯讀，顯示此 API 的 C 語言聲明。



點擊“確定”後，添加此 API 的 BASIC 介面；點擊“取消”，取消此次編輯。

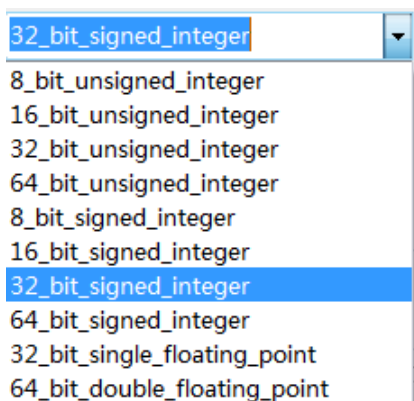
- 添加結構體

添加結構體需配置如下參數：

- 名稱：結構體的名字，BASIC 關鍵字，不可與已存在的 BASIC 關鍵字重複
- 對齊模式：可選擇 1，2，4，8，16。默認為 8。
- 成員：結構體中的成員，左邊列表列出已定義的成員名，選擇後，右邊為當前成員的具體資訊；可對成員進行添加/刪除/上移/下移的操作。
- 當前成員：顯示/編輯當前成員的資訊，包含名稱，類型，資料類型和傳遞方式。  
 名稱：除第一個參數名默認為返回值（唯讀）外，其他函數名只能為字元，數位和底線。  
 類型：此參數的類型，分為 Numeric（數位），Array（陣列），String（字串）和結構體（Struct）

資料類型：根據所選類型而變化。

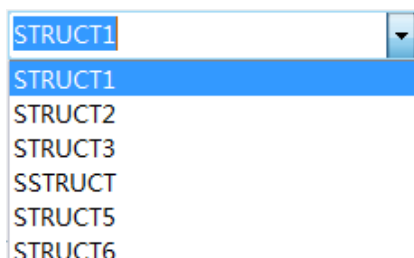
當類型為 Numeric 或者 Array 時，資料類型有如下幾類：



當類型為 **String** 時，沒有資料類型，需要填寫元素個數（同陣列類型），即字串的長度

元素个数:

當類型為 **Struct** 時，資料類型將列出已添加支援的結構體類型，如下：

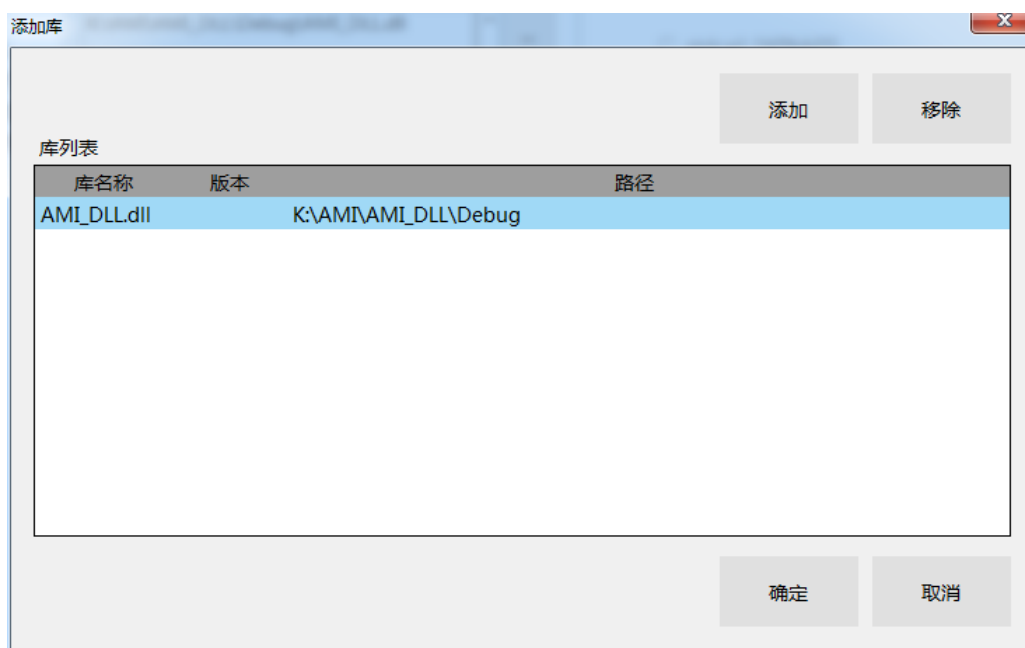


常量：勾選表示參數為常量

- 結構體原型：唯讀，顯示此結構體的 C 語言聲明。

點擊“確定”後，添加此結構體的 **BASIC** 介面；點擊“取消”，取消此次編輯。

- 添加庫



添加庫頁面將會列出目前已添加的庫列表，可添加或移除已有的庫。

每一行會顯示此庫的名稱，版本（若存在）以及庫路徑。  
點擊“確定”將確定本次編輯；點擊“取消”，將取消此次編輯。

## 9.10 選項

選項主要為配置 Studio 的工具的可選設置。

目前分為以下三類：

常規：關於 Studio 和工具的常用可選設置。

編輯器：關於編輯器的常用可選設置。

項目：關於專案的常用可選設置。



### ● 常規

主要分為兩項

#### 1) 啟動模式：Studio 開啟後的啟動模式

選項	含義
上次關閉的模式	Studio 會記錄上次關閉時的模式，打開會設置為關閉時的模式。 若關閉時為斷線模式，連線後會自動切換為監控模式
斷線模式	Studio 打開後自動設置為斷線模式
監控模式	Studio 打開後自動設置為監控模式
同步模式	用最後使用的專案與控制器建立同步連接
啟動後提示	Studio 每次啟動後跳出對話方塊，選擇操作模式

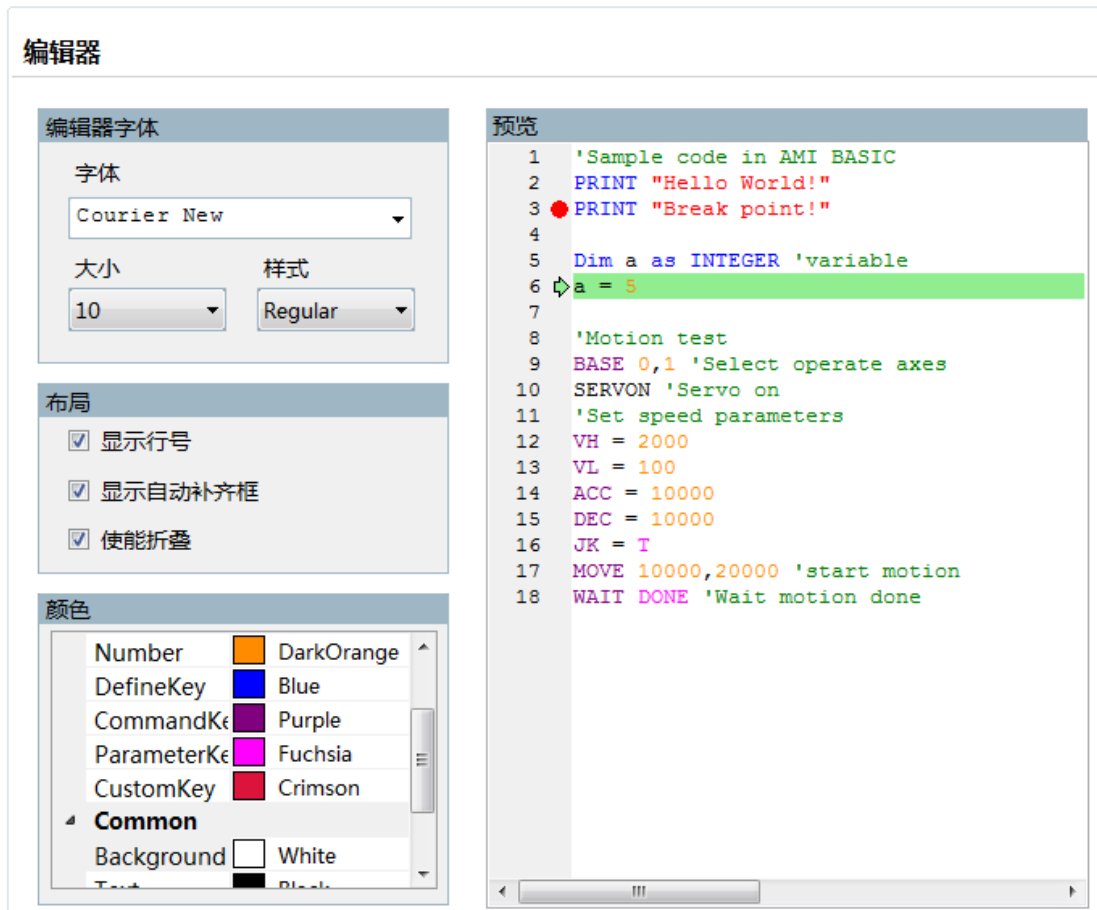
#### 2) 刷新頻率

對 Studio 中的工具設置刷新頻率。

注意：刷新頻率過高，可能會增加 Studio 的負荷，影響整體使用效果。因此，建議在非必須的情況下，不要設置過高的刷新頻率。

- 編輯器

對編輯器的字體，關鍵字的顏色等進行設置



左邊為設置，右邊為預覽圖。

設置部分分為三大塊：

- 1) 字體：包含字體名稱，大小和樣式
- 2) 佈局：是否顯示行號
- 3) 顏色：關鍵字的顏色

目前分為以下幾類：

Comment：注釋文字

String：字串

Number：數字

CommandKey：MAS BASIC 命令相關以及系統相關關鍵字

ParameterKey：MAS BASIC 屬性/狀態相關關鍵字

CustomKey：使用者自訂的庫函數中的相關關鍵字（API 和結構體）

Background：背景色

Text：文字（非關鍵字）顏色

Breakpoint：背景色

DebugStep：調試單步高亮行顏色

- 項目

專案相關的常用配置

1) 同步

配置專案比較/合併工具：

同步時，處理差異，點擊比較項時，使用配置的比較工具進行兩者檔的比較

目前 Studio 默認下載專案時列出自動運行的程式，控制器類型與配置不一致時會跳出對話方塊提示。

2) 備份

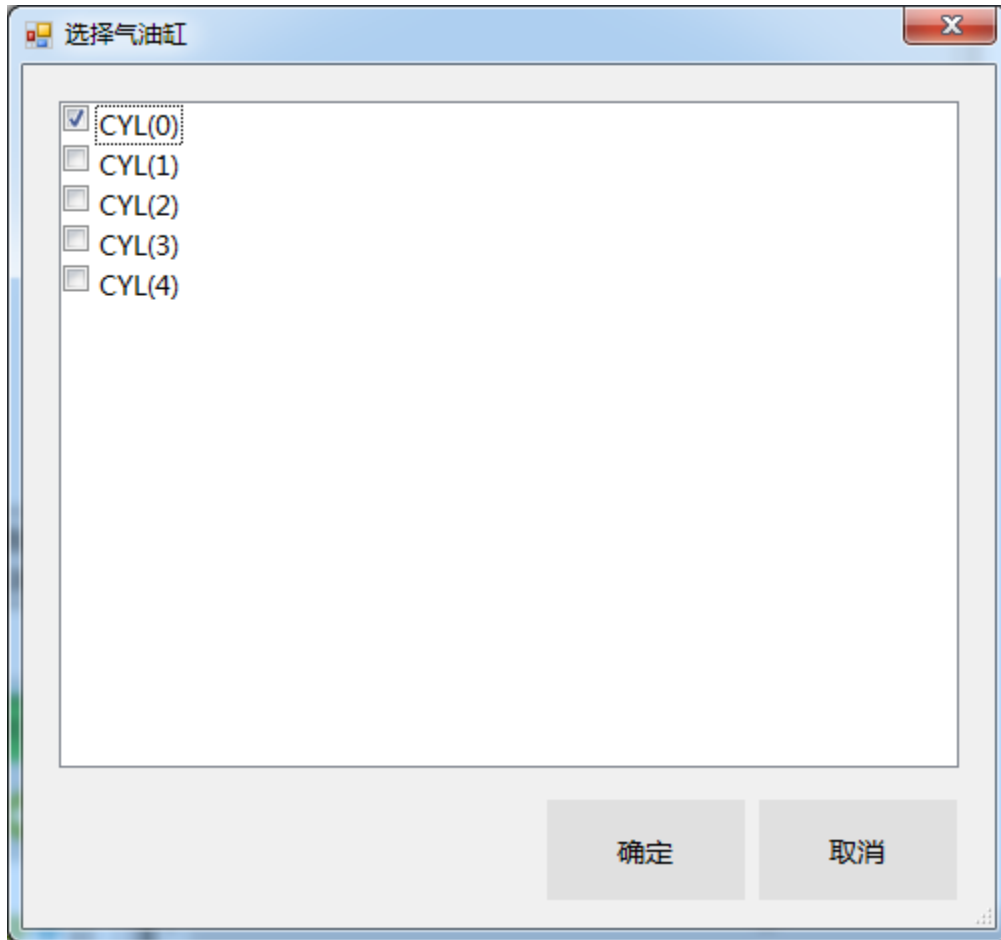
從備份恢復項目時，Studio 會預設恢復工具配置

注：目前桌面佈局暫時不可恢復

## 9.11 氣油缸測試

為測試氣油缸相關功能，添加氣油缸測試工具。

當配置好氣油缸後，可點擊【選擇氣油缸】，對相應氣油缸進行功能測試。



氣油缸測試介面如下：

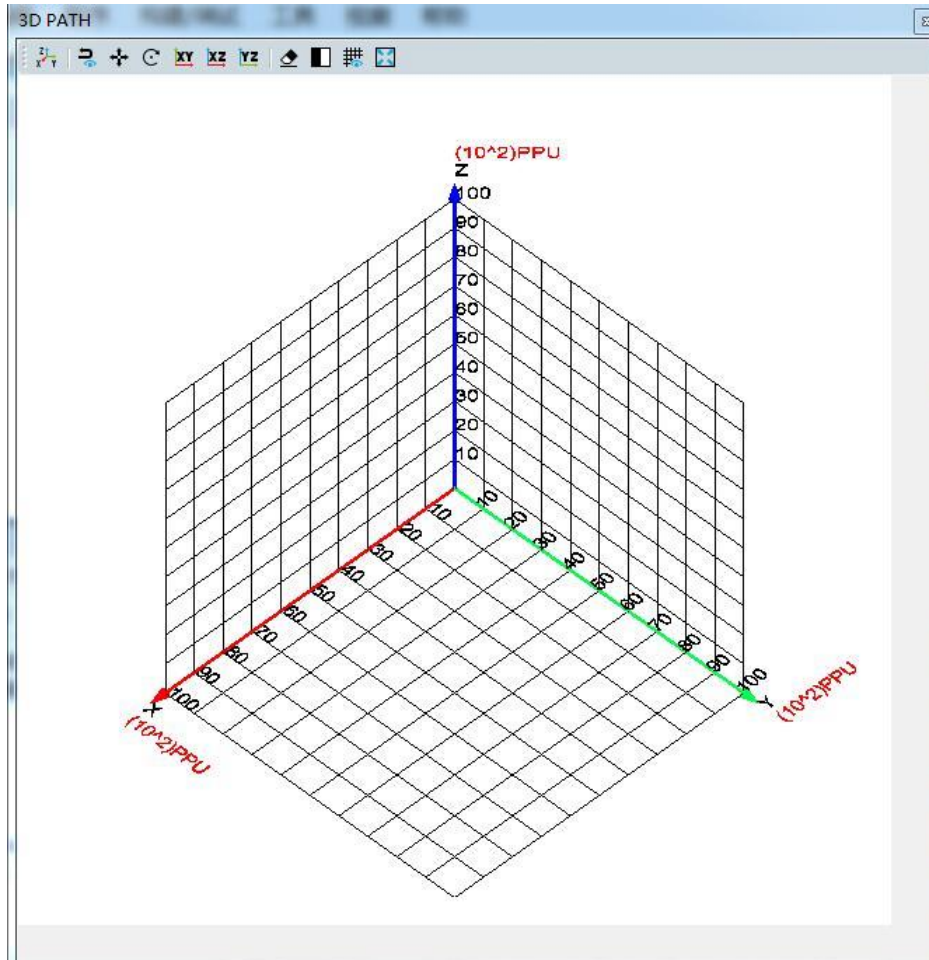


針對某一個氣油缸，可進行前進/後退/停止/復位操作。通過搭配當前氣油缸狀態以及對應

DO/DI 狀態，測試此氣油缸的功能是否正常。

## 9.12 3D 軌跡

3D 軌跡用於顯示 MAS 控制器指定軸的運動軌跡。



### 9.12.1 主要功能

- 3D 視角、2D 視角
- 平移
- 旋轉
- 放大縮小
- 刻度及網格顯示
- 軌跡顯示
- 軌跡清除
- 背景色切換
- 網格及刻度是否顯示按鍵
- 配置視窗

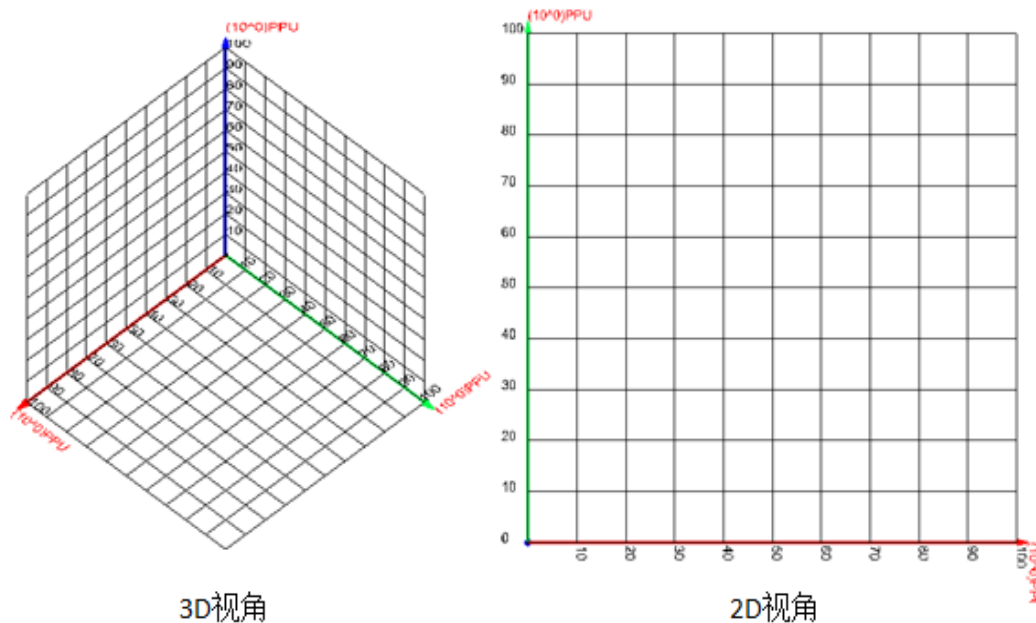
## 9.12.2 功能介紹



圖 2-1

- 視角切換

初始視角預設為 3D 視角。如圖 2-1，“初始狀態”按鈕為重置為初始狀態，並切換為 3D 視角；“XY 平面”、“XZ 平面”、“YZ 平面”按鈕為對應的 2D 視角。



- 平移

如圖 2-1，點擊“平移按鈕”，再用滑鼠在視窗內拖動，即可平移畫面。

- 旋轉

如圖 2-1，點擊“旋轉按鈕”，在視窗內點擊滑鼠，會出現一個橙色的圓，如果滑鼠點擊位置在圓內，左右拖動滑鼠控制畫面繞 X 軸旋轉，上下拖動滑鼠控制畫面繞 Y 軸旋轉；如果滑鼠點擊位置在圓外，上下拖動滑鼠控制畫面繞 Z 軸旋轉。

**注：2D 視角沒有旋轉功能。**

- 放大/縮小

在視窗內滾動滑鼠滾輪，畫面會以滑鼠位置為中心進行縮放，滾輪上滾放大，滾輪下滾縮小。

- 刻度及網格顯示

預設坐標系顯示 10 個刻度及網格，滾動滑鼠放大畫面到一定程度，會分出更小的次網格及刻度，最多 250 個刻度及網格。

- 軌跡顯示

即時讀取 DPOS 值，並在畫面中顯示軌跡。

- 清除軌跡

如圖 2-1，點擊工具列的“清除軌跡”按鈕，可以清除當前畫面上的軌跡。

- 背景色切換

如圖 2-1，點擊工具列的“切換背景”按鈕，可以使畫面黑\白兩種背景切換

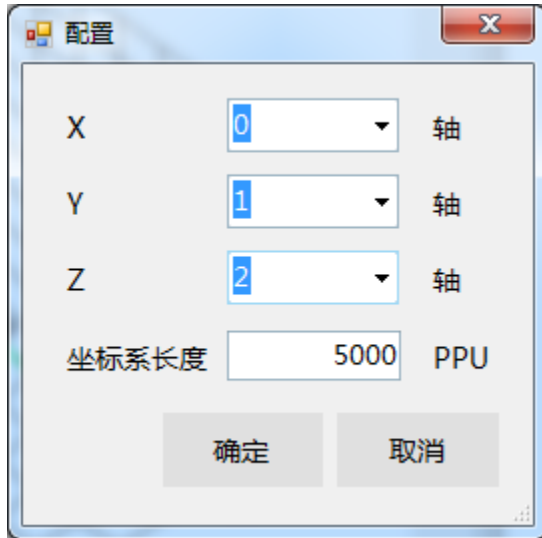


- 網格及刻度是否顯示按鍵

如圖 2-1，點擊工具列“網格顯示\隱藏”按鍵，可以控制網格及刻度是否可見

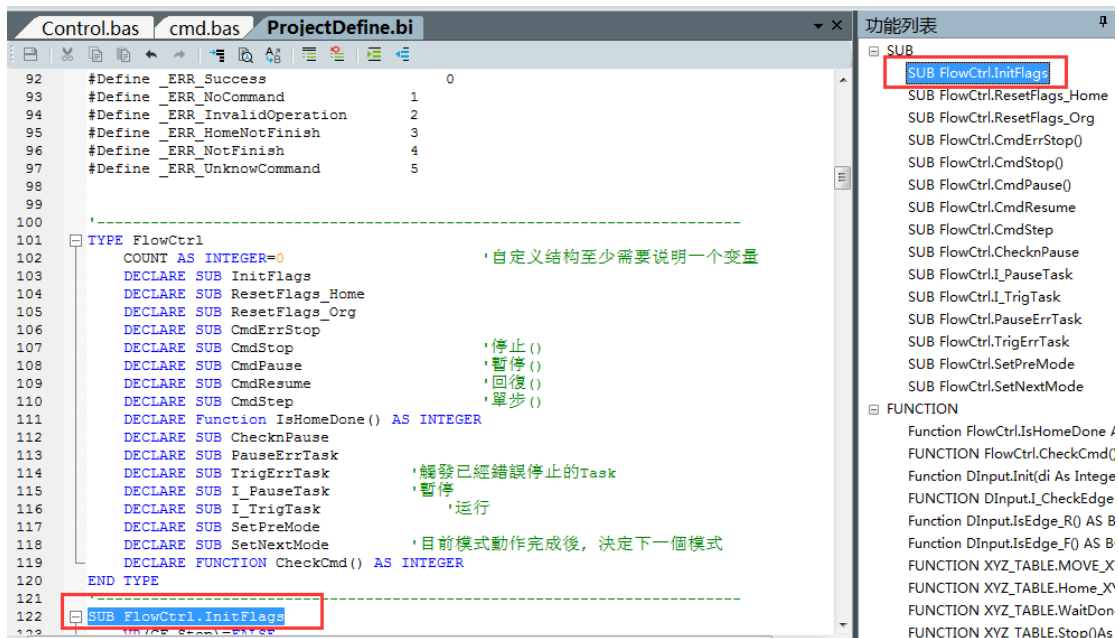
- 配置視窗

如圖 2-1，點擊工具列“配置”按鍵，彈出配置視窗，配置視窗可設置 XYZ 對應的軸、坐標系長度



## 9.13 功能列表

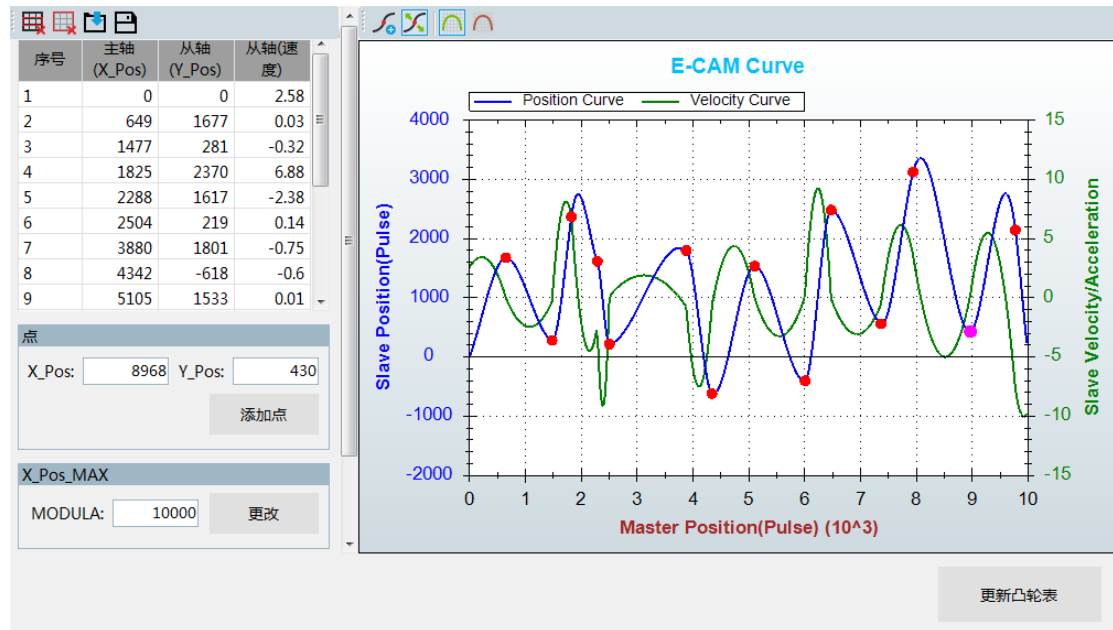
為使用者可比較快速查看/定位自訂函數 (SUB/FUNCTION) /類 (TYPE)，提供功能列表。列表中會列出當前活動.bas/ProjectDefine.bi 中的所有 SUB/FUNCTION/TYPE。點擊功能列表中具體 SUB/FUNCTION/TYPE 節點，將定位到對應的 SUB/FUNCTION/TYPE。



## 9.14 凸輪編輯器

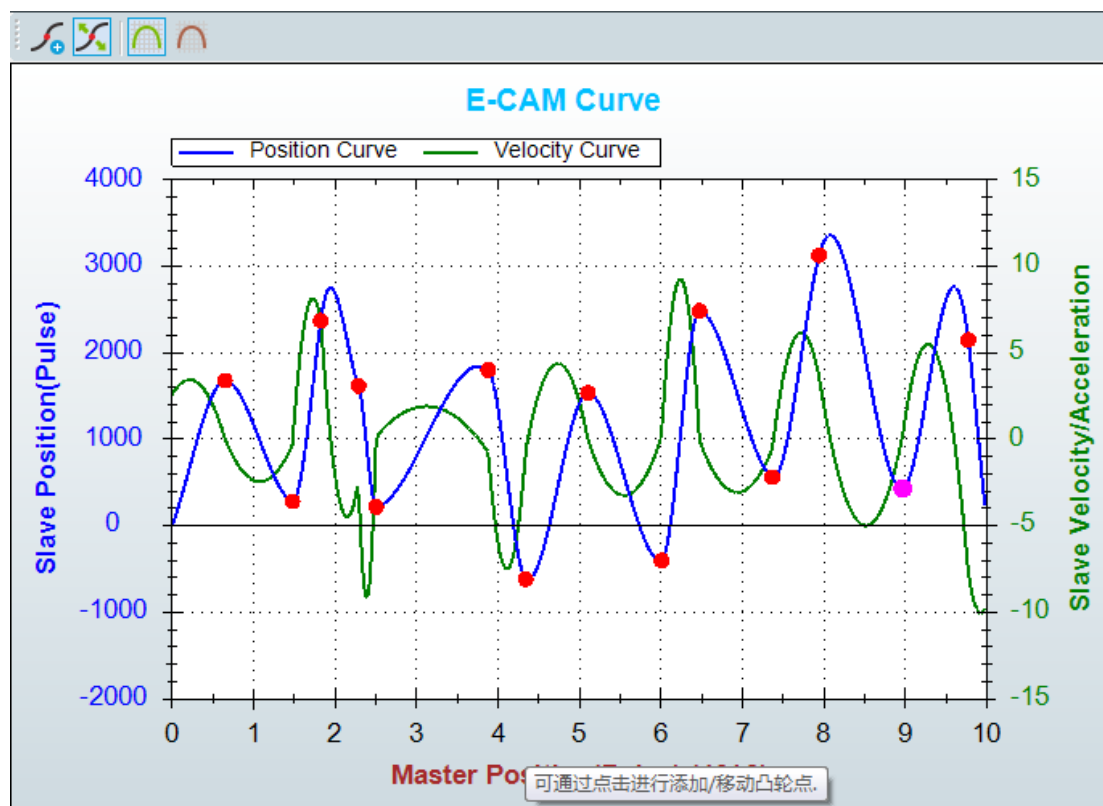
為更好地支援凸輪功能，提供凸輪編輯器。使用者可在此工具中編輯凸輪曲線，並可設定關聯的 VR 起始位址，使凸輪點與 VR 進行關聯，在.bas 中使用凸輪指令實現凸輪功能。

凸輪編輯器介面如下：




大致可分為以下兩大區域。


## 9.14.1 曲線編輯/顯示區



- 滑鼠操作





當滑鼠操作為  (添加點) 時，可直接在曲線編輯/顯示區通過滑鼠點擊添加凸輪點；

當滑鼠操作為  (選中點) 時，可選中相應的凸輪點拖拉對曲線進行更改/微調；

- 顯示曲線



當勾選  時，圖形顯示區將增加速度曲線的顯示；反之則取消速度曲線的顯示；


當勾選  時，圖形顯示區將增加加速度曲線的顯示；反之則取消加速度曲線的顯示；


## 9.14.2 凸輪表相關配置/操作

- 凸輪列表及相關操作


序号	主轴 (X_Pos)	从轴 (Y_Pos)	从轴(速 度)
1	0	0	2.58
2	649	1677	0.03
3	1477	281	-0.32
4	1825	2370	6.88
5	2288	1617	-2.38
6	2504	219	0.14
7	3880	1801	-0.75
8	4342	-618	-0.6
9	5105	1533	0.01

座標為已添加的凸輪列表，按照主軸（X\_Pos）的值依次排列。選中相應列，對應凸輪點會高亮顯示。

選中對應行（第一行/最後一行除外），點擊 ，可刪除對應的凸輪點

點擊 ，將清除所有凸輪點（起始點（第一行）/終點（最後一行）除外）。清除後，將自動切換到“添加點”模式

點擊 ，可導入之前保存的凸輪列表（支持.bin/.csv 兩種方式）

點擊 ，可保存凸輪列表（支援.bin/.csv 兩種方式）。

- 點

点

X\_Pos:  Y\_Pos:

添加点

除了通過滑鼠點擊方式添加凸輪點外，用戶也可直接在此編輯 X\_Pos（主軸）和 Y\_Pos（從軸）的位置，點擊【添加點】來添加凸輪點。

- 凸輪表關聯 VR

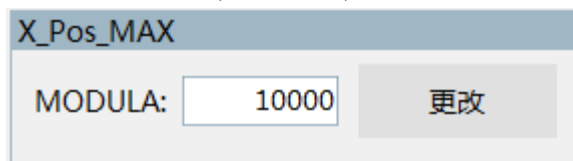
凸轮表关联VR

起始VR:  到

設置起始 VR，編輯完凸輪曲線，點擊【應用凸輪表】會將凸輪點資訊保存到 VR。與凸輪指令一致，保存的順序為：

VR\_Start:  
 - VR(VR\_Start) --- x pos of first point  
 - VR(VR\_Start +1) --- y pos of first point  
 - VR(VR\_Start +2) --- x pos of second point  
 - VR(VR\_Start +3) --- y pos of second point  
 - .....

- X\_Pos\_MAX(MODULA)



MODULA 含義為凸輪主軸轉動一周的 Pulse 數，即對應橫坐標的最大值。可按照實際情況進行更改。

關閉表單後，再次打開，滑鼠操作將預設為【選中點】。

注：若編輯完，未點擊【應用凸輪表】，關閉表單，編輯實際不會生效，上次編輯效果也不會保存。