



快速上手



330 桃園市同德十一街 58 號 10 樓之 2

Tel: 03.358.6008 Fax: 03.358.6009

E-mail: info@montrol.com.tw

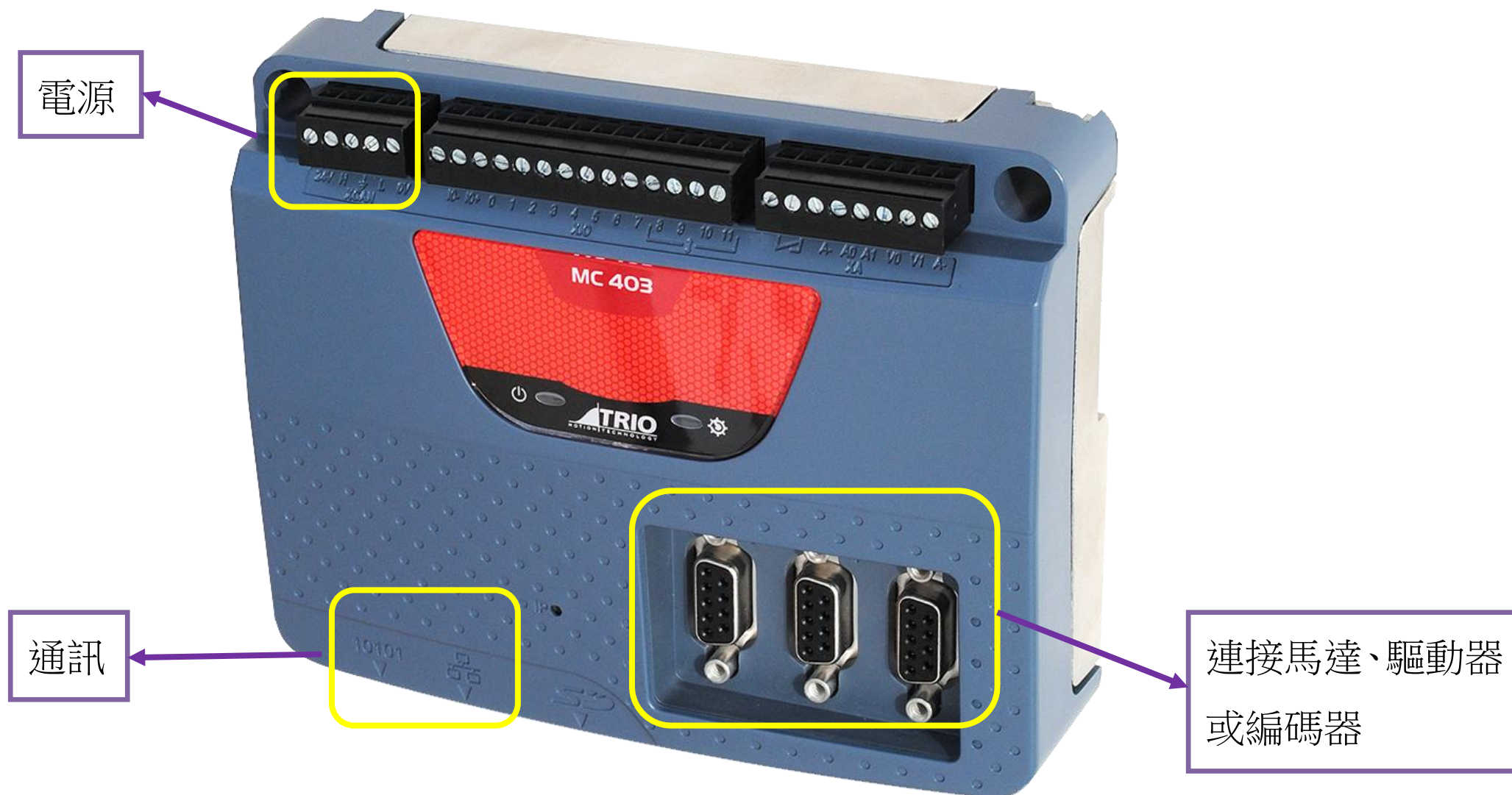
敏石系統有限公司

www.montrol.com.tw

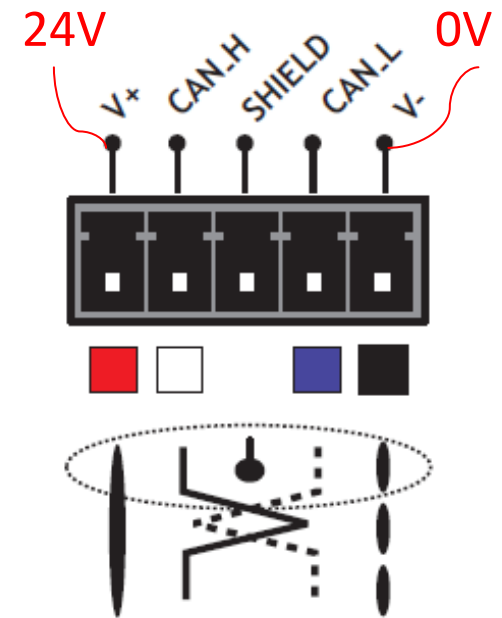
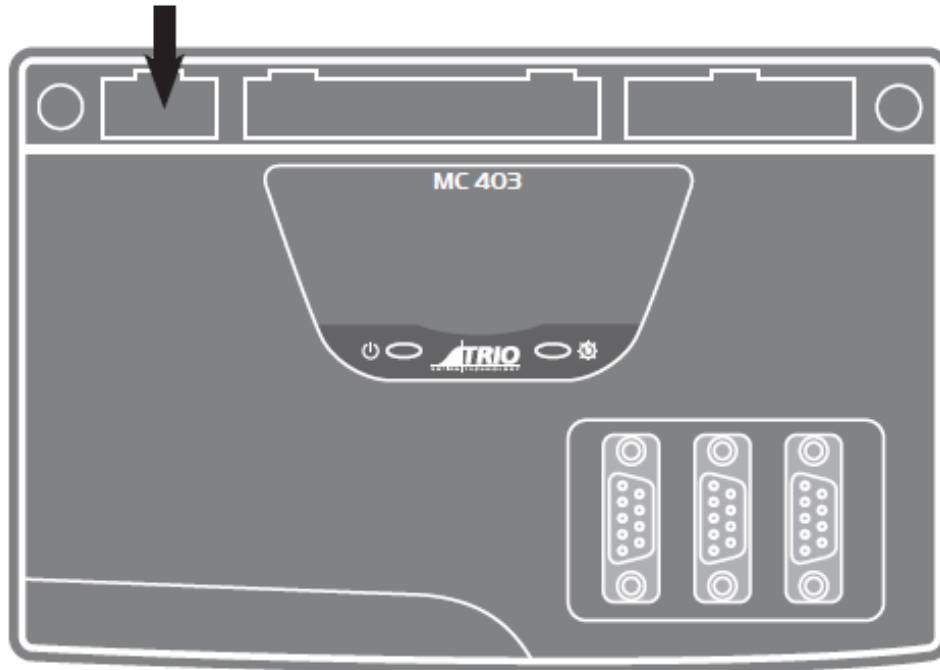


■ 目錄

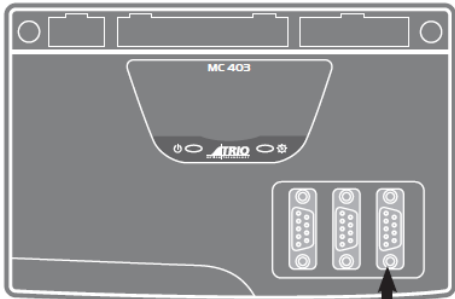
硬體基本接線：電源、馬達軸、通訊	3
軟體操作	
建立通訊	7
軟體介面：Controller 視窗、Axis Parameters 視窗、I/O 視窗、Oscilloscope 視窗	13
終端機	17
程式編輯	19
程式指令基礎介紹：UNITS 用法	25
主題一、多軸同時開始運動	26
主題二、多軸補間運動	27
主題三、兩軸運動-畫圓	28
主題四、兩軸運動-畫多個圓弧 & MERGE	29
硬體 I/O 接線	30
主題五、 迴圈 LOOP (一) WHILE.. WEND	34
迴圈 LOOP (二) REPEAT..UNTIL	35
主題六、 輸入控制持續運動(一) FORWARD & REVERSE	36
輸入控制持續運動(二) FWD JOG & REV JOG	37
主題七、I/O 控制其他程式碼執行 PROC STATUS	38
主題八、原點復歸(DATUM)	39
主題九、副程式	40
指令說明	41



5-WAY CONNECTOR

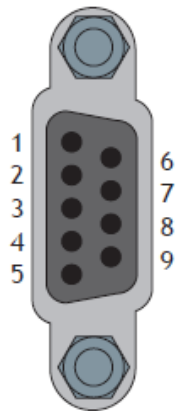


MC403 PULSE OUTPUTS / ENCODER INPUTS



Axis 0 1 2

程式說明部分會以軸 0 與 2 為例

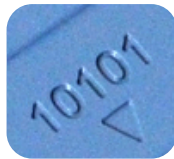
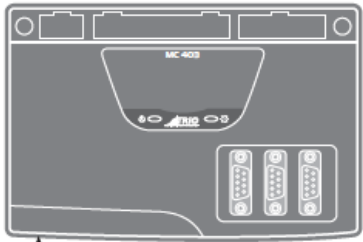


Pin	Function	Pulse & Direction	Absolute Encoder **
1	Enc. A	Step+	Clock+
2	Enc. /A	Step-	Clock-
3	Enc. B	Direction+	N/C
4	Enc. /B	Direction-	N/C
5	0V Encoder	0V Pulse+direction	0V Encoder
6	Enc. Z	Enable+	Data+
7	Enc. /Z	Enable-	Data-
8	5V *	5V*	5V*
9	N/C	N/C	N/C

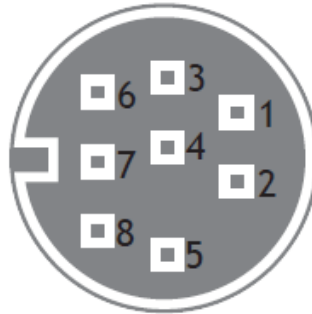
*5V supply is limited to 150mA (shared with serial port)

**Not available on axes 0 and 1 of the MC403-Z

SERIAL CONNECTIONS

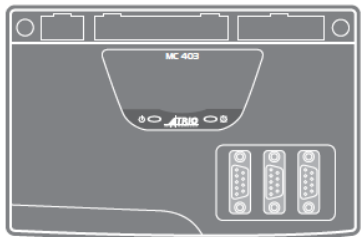


Serial Connector



Pin	Function	Note
1	RS485 Data In A Rx+	Serial Port #2
2	RS485 Data In B Rx-	
3	RS232 Transmit	Serial Port #1
4	0V Serial	
5	RS232 Receive	Serial Port #1
6	5V	
7	RS485 Data Out Z Tx-	Serial Port #2
8	RS485 Data Out Y Tx+	

RJ45 CONNECTOR

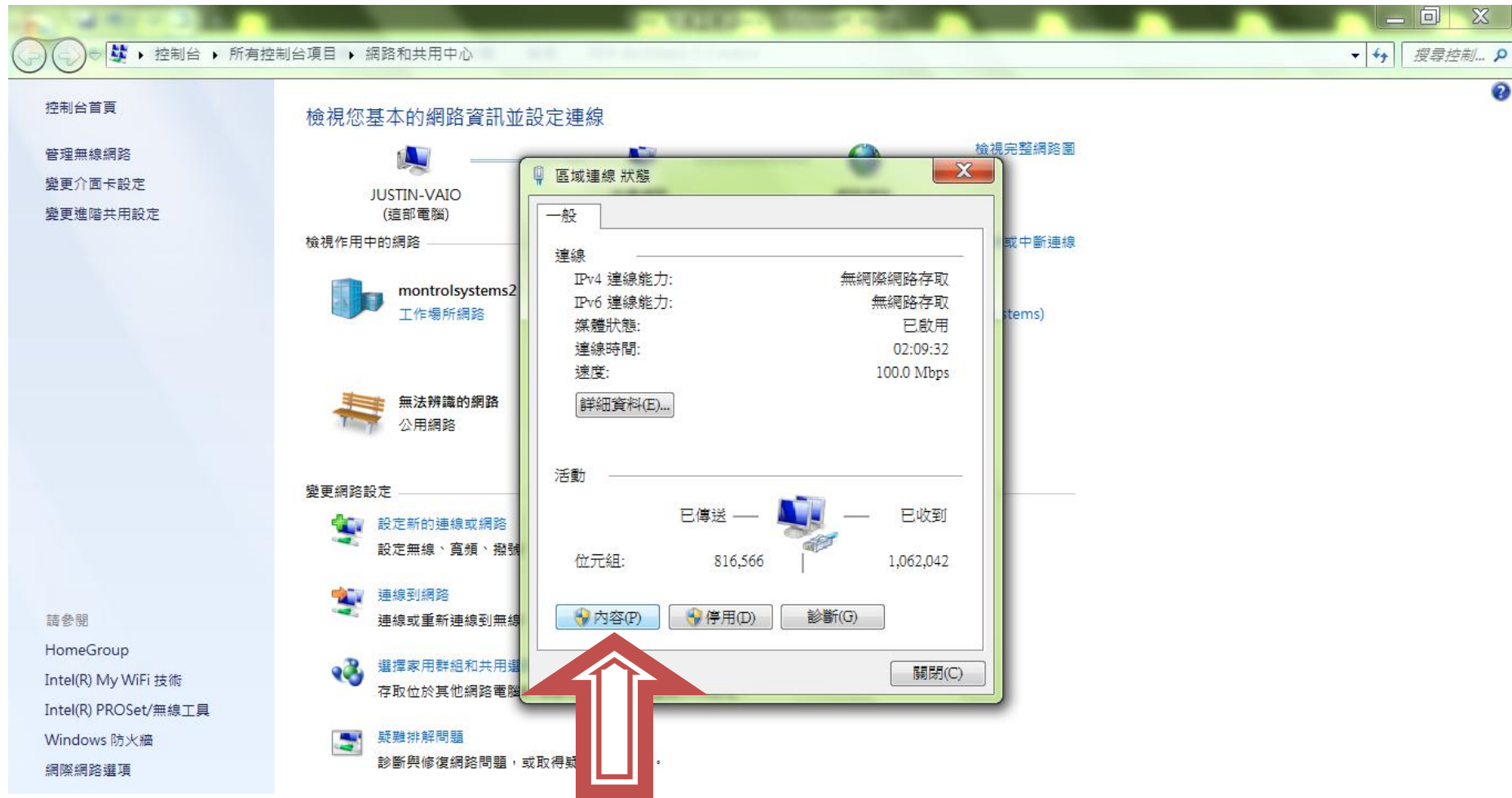


建議： Ethernet 網路線請使用 CAT5 以上等級

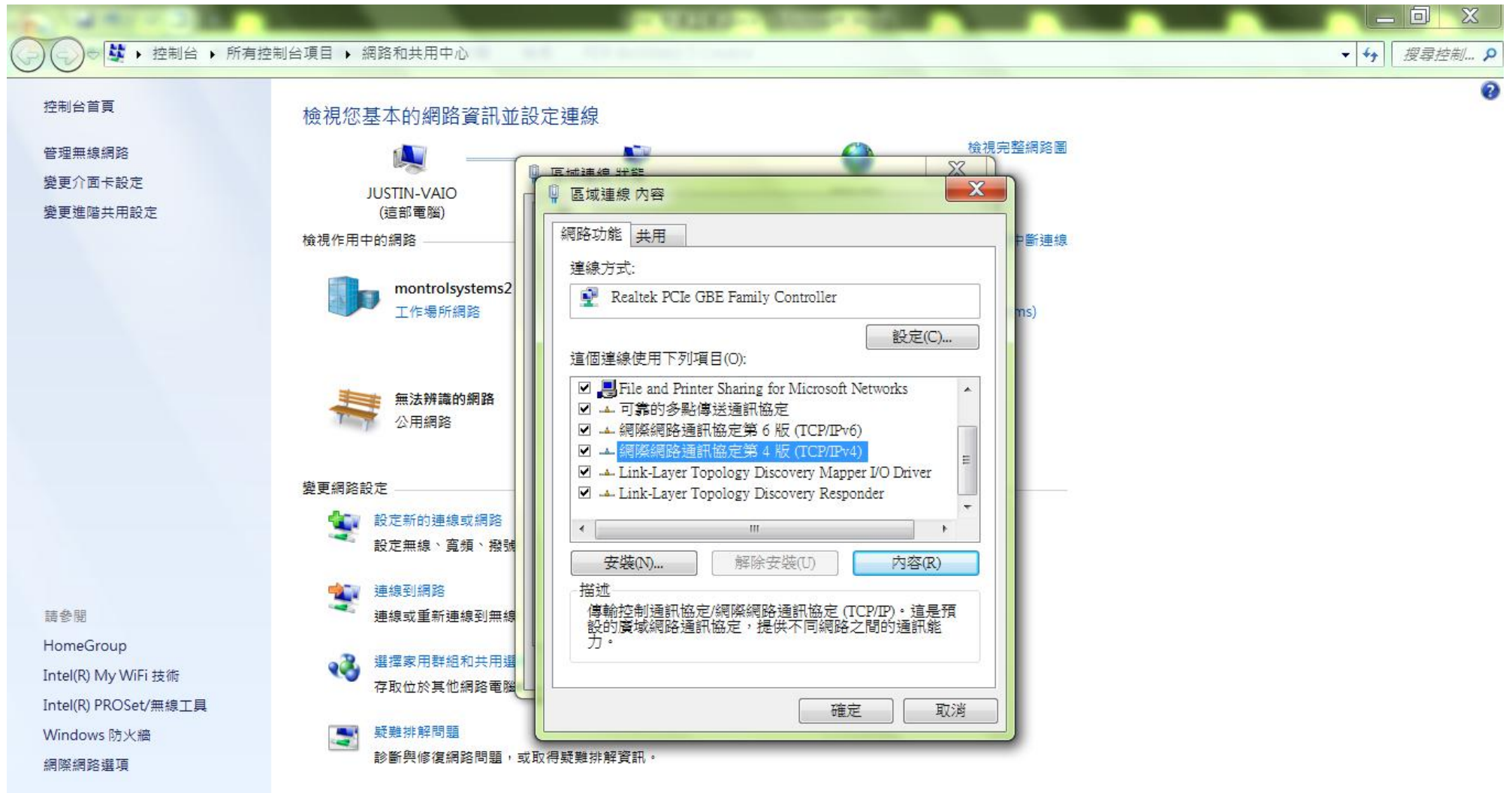
1. 開啟 **網路和共用中心**，找尋 **區域連線**，點擊。



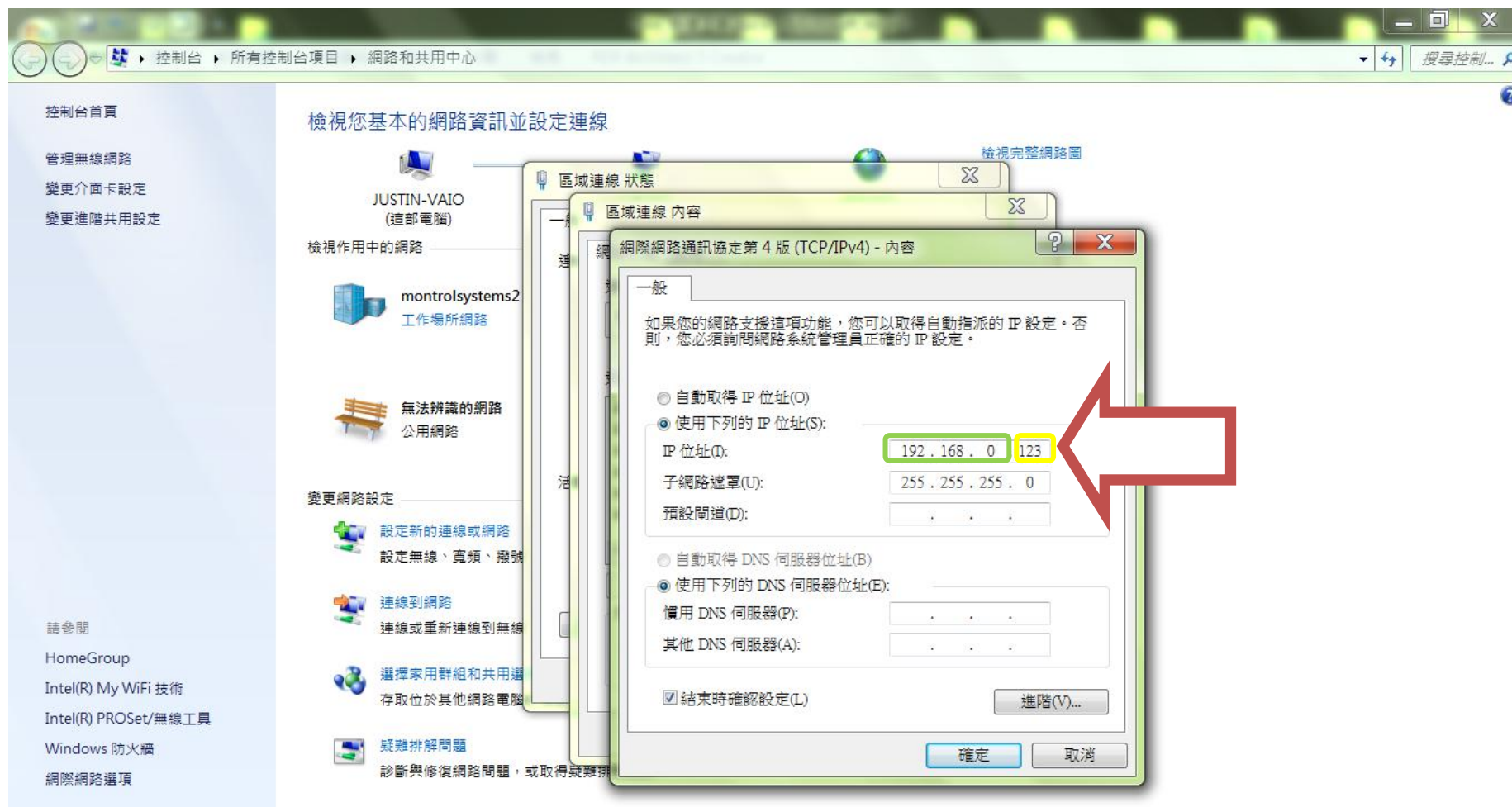
2. 點擊內容。





3. 找尋網際網路通訊協定第 4 版(TCP/IPv4) -> 按內容。

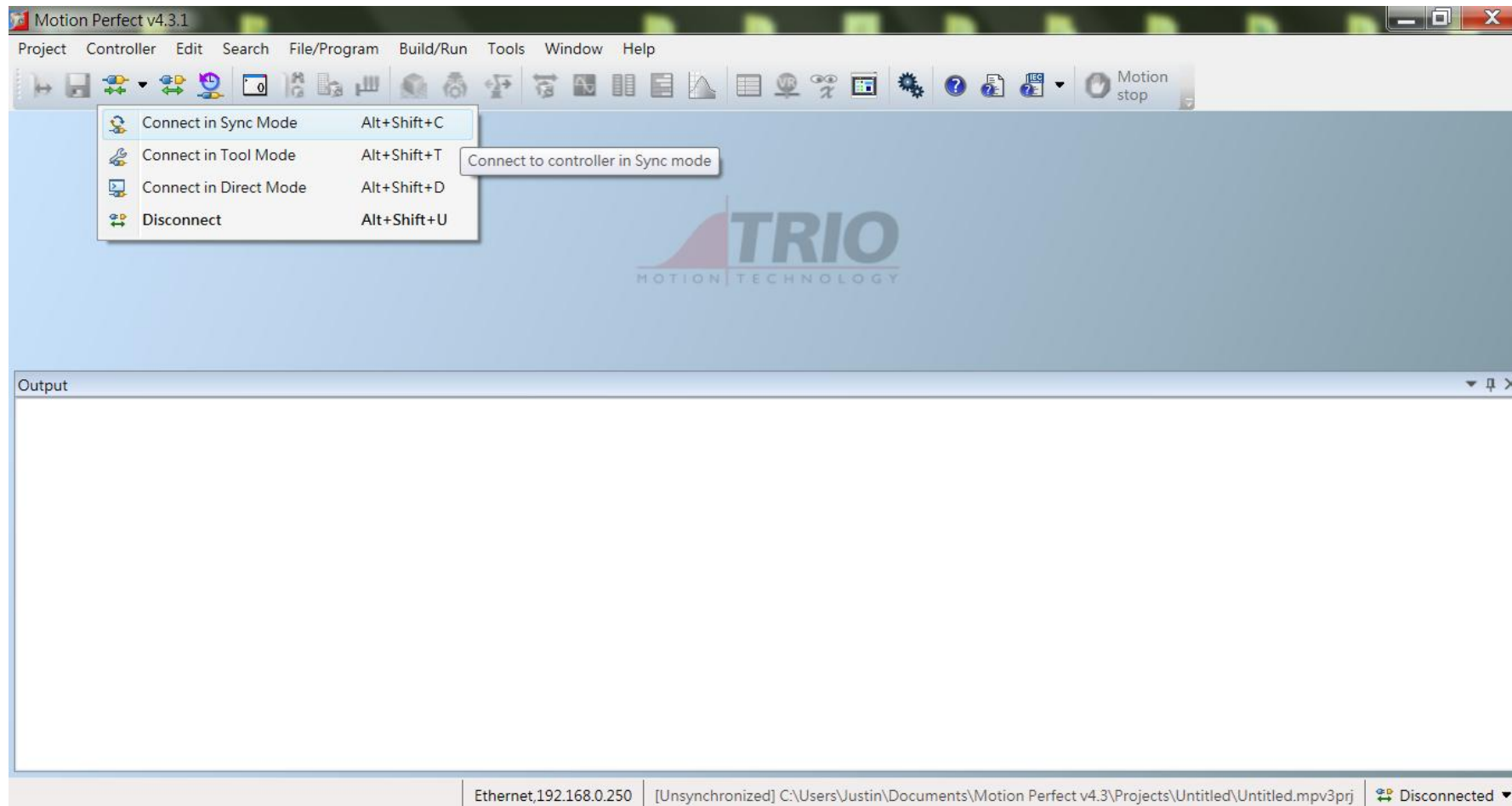


4. 選項修改為**使用下列的 IP 位址**，設定電腦部份 IP 位址，前三碼 **192.168.0** 與軟體上 Controller IP Address 相同(詳見第 6 點)，末一碼 **123** 不可相同(軟體預設 250)，並按確定。

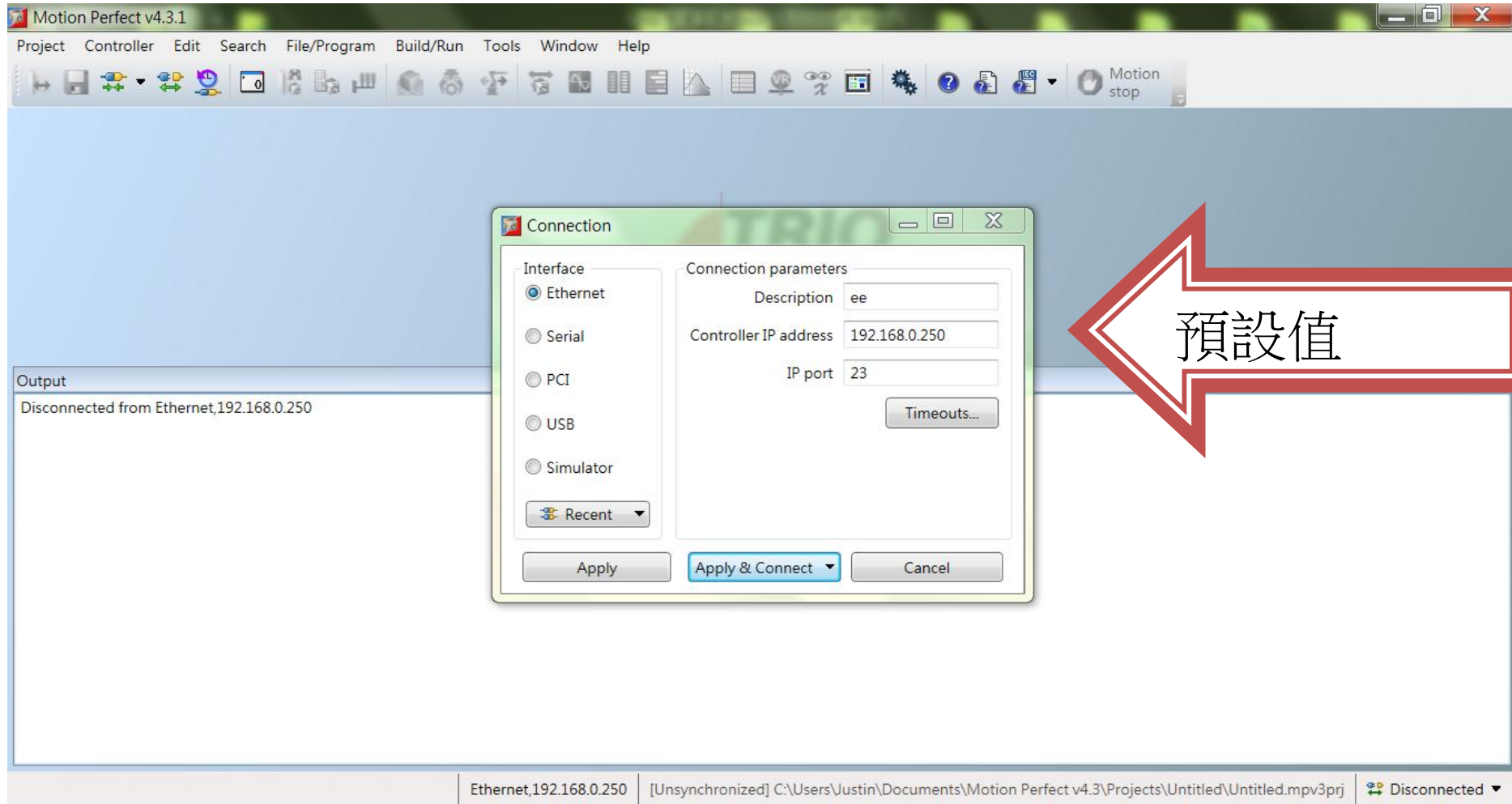




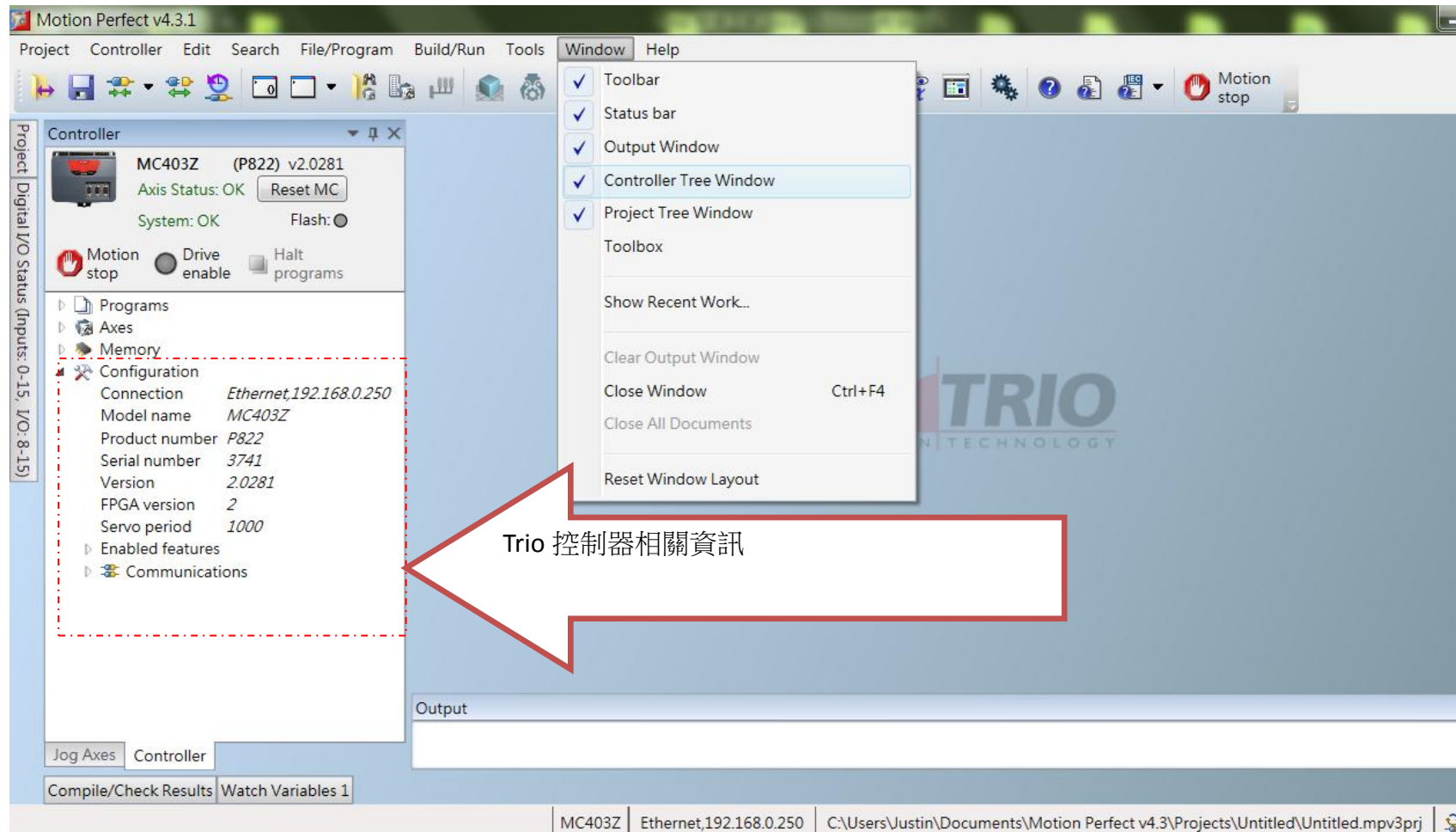
5. 滑鼠雙擊圖示 ，開啟 Motion Perfect 軟體，點選工具選單 Controller 或圖示  -> Connect in Sync Mode，與機台連線。



6. 選擇連線方式(以 Ethernet 為例)，第一次連線須做設定(之後軟體會自動記憶先前的連線方式)，並點選 **Apply & Connect** 完成連線。



1. Controller 視窗：點選功能選單 Window -> 勾選 Controller Tree Window。





2. Axis Parameters 視窗：點擊圖示 或點選功能選單 Tools ->開啟 Axis Parameters，可查看運動軸的參數設定及回傳值。

Motion Perfect v4.3.1

Project Controller Edit Search File/Program Build/Run Tools Window Help

Axis Parameters

Select axes 1.0 (Type text to search for)

Parameter	Axis (0)	Axis (1)	Axis (2)
ATYPE	Step+Dir	Step+Dir	Step+Dir
UNITS	1.0	1.0	1.0
▲ Gains			
P_GAIN	1.0	1.0	1.0
I_GAIN	0.0	0.0	0.0
D_GAIN	0.0	0.0	0.0
OV_GAIN	0.0	0.0	0.0
VFF_GAIN	0.0	0.0	0.0
▲ Velocity profile			
ACCEL	10000.0	10000.0	10000.0
DECEL	10000.0	10000.0	10000.0
SPEED	1000.0	1000.0	1000.0
CREEP	100.0	100.0	100.0
MERGE	0	0	0
SRAMP	0	0	0
MSPEED	0.0	0.0	0.0
VP_SPEED	0.0	0.0	0.0
▲ Limits			
DATUM_IN	-1	-1	-1
FE_LIMIT	20000.0	20000.0	20000.0
FE_RANGE	10000.0	10000.0	10000.0
FHOLD_IN	-1	-1	-1
FWD_IN	-1	-1	-1
REP_DIST	200000000000.0	200000000000.0	200000000000.0

Axis Parameters Jog Axes Controller

Compile/Check Results Watch Variables 1

MC403Z Ethernet,192.168.0.250 C:\Users\Justin\Documents\Motion Perfect v4.3\Projects\Untitled\Untitled.mpv3prj Sync mode



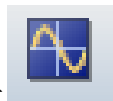
3. I/O 視窗：點擊圖示 或點選功能選單 Tools ->開啟 Digital I/O Status，可查看數位 I/O 狀態。

The screenshot displays the Motion Perfect v4.3.1 software interface. The main window is titled "Digital I/O Status (Inputs: 0-15, I/O: 8-15)". The interface includes a menu bar (Project, Controller, Edit, Search, File/Program, Build/Run, Tools, Window, Help) and a toolbar with various icons. A "Motion stop" button is visible in the toolbar. The main area shows a "Digital I/O status" window with two tables:

Built-In Inputs	
In	Description
0	
1	
2	
3	
4	
5	
6	
7	

Built-In I/Os	
I/O	Description
8	
9	
10	
11	
12	
13	
14	
15	


The interface also features a sidebar on the left with "Project", "Digital I/O Status (Inputs: 0-15, I/O: 8-15)", and "Controller" tabs. On the right, there are tabs for "Oscilloscope", "Channel #0", "Jog Axes", and "Axis Parameters". The status bar at the bottom shows "MC403Z", "Ethernet,192.168.0.250", "C:\Users\Justin\Documents\Motion Perfect v4.3\Projects\Untitled\Untitled.mpv3prj", and "Sync mode".

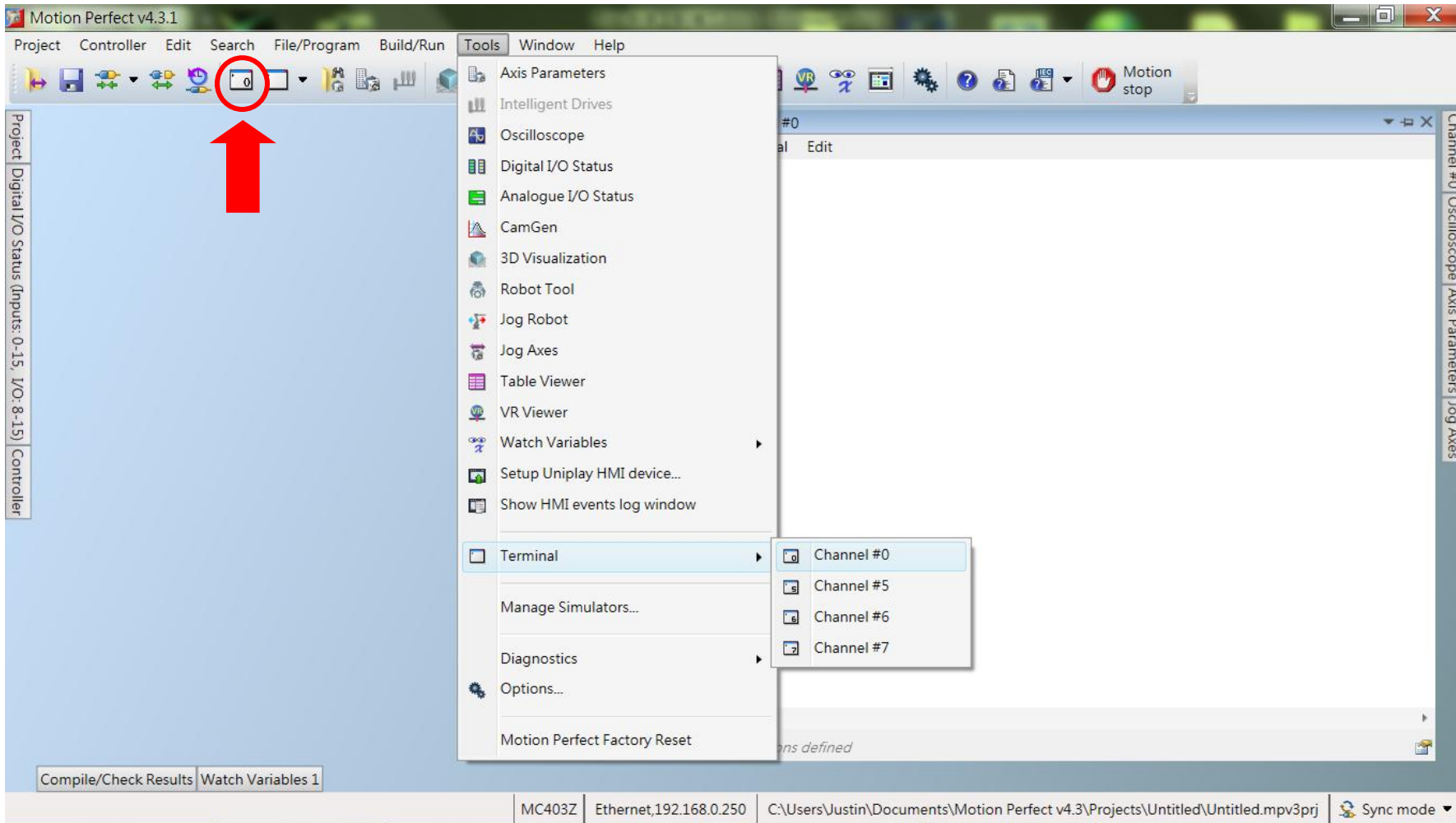


4. Oscilloscope 視窗：點擊圖示 或點選功能選單 Tools ->開啟 Oscilloscope，可查看運動軌跡或波形。

The screenshot displays the Motion Perfect v4.3.1 software interface. The main window is titled "Motion Perfect v4.3.1" and features a menu bar (Project, Controller, Edit, Search, File/Program, Build/Run, Tools, Window, Help) and a toolbar. The "Tools" menu is open, highlighting the "Oscilloscope" option. The Oscilloscope window is active, showing a configuration panel on the left and a grid display on the right. The configuration panel includes settings for multiple channels (Axis 0 and Axis 2) with variables like DPOS and DEMAND_SPEED. The grid display shows "Ch.0 No data" and "Ch.1 No data". The bottom status bar indicates the system is in "Sync mode" and provides connection details for MC403Z.

MC403Z | Ethernet,192.168.0.250 | C:\Users\Justin\Documents\Motion Perfect v4.3\Projects\Untitled\Untitled.mpv3prj | Sync mode

點擊圖示  或點選功能選單 Tools -> Terminal 開啟



軸參數預設值 UNITS=1; ACCEL=10000; DECEL=10000; SPEED=1000

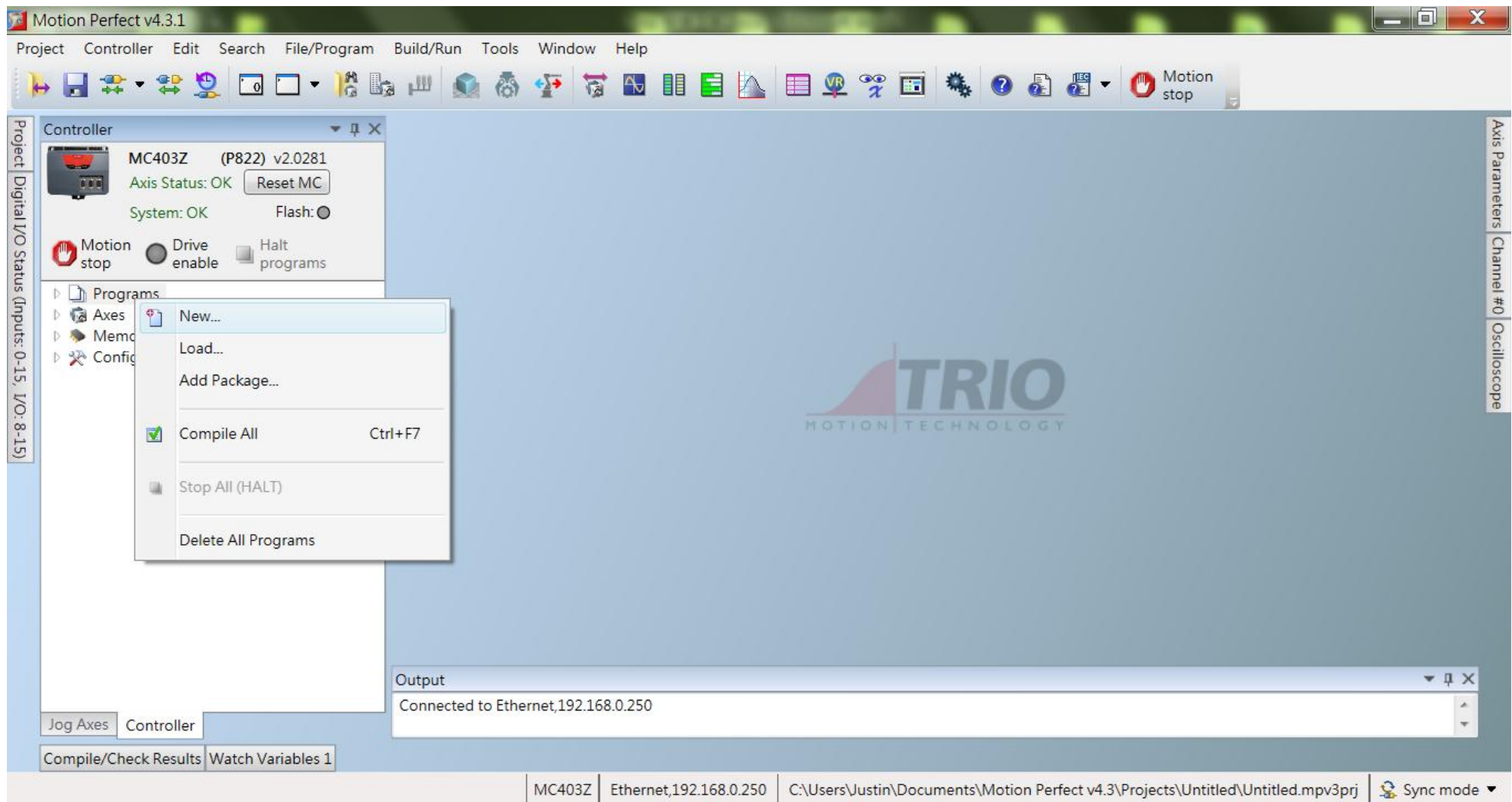
以步進馬達解析度 25000 為例，UNITS=25000*16; ACCEL=10; DECEL=10; SPEED=1，下指令 MOVE(1)可使馬達 1 秒內轉 1 圈

The screenshot displays the Motion Perfect v4.3.1 software interface. The main window is divided into several sections:

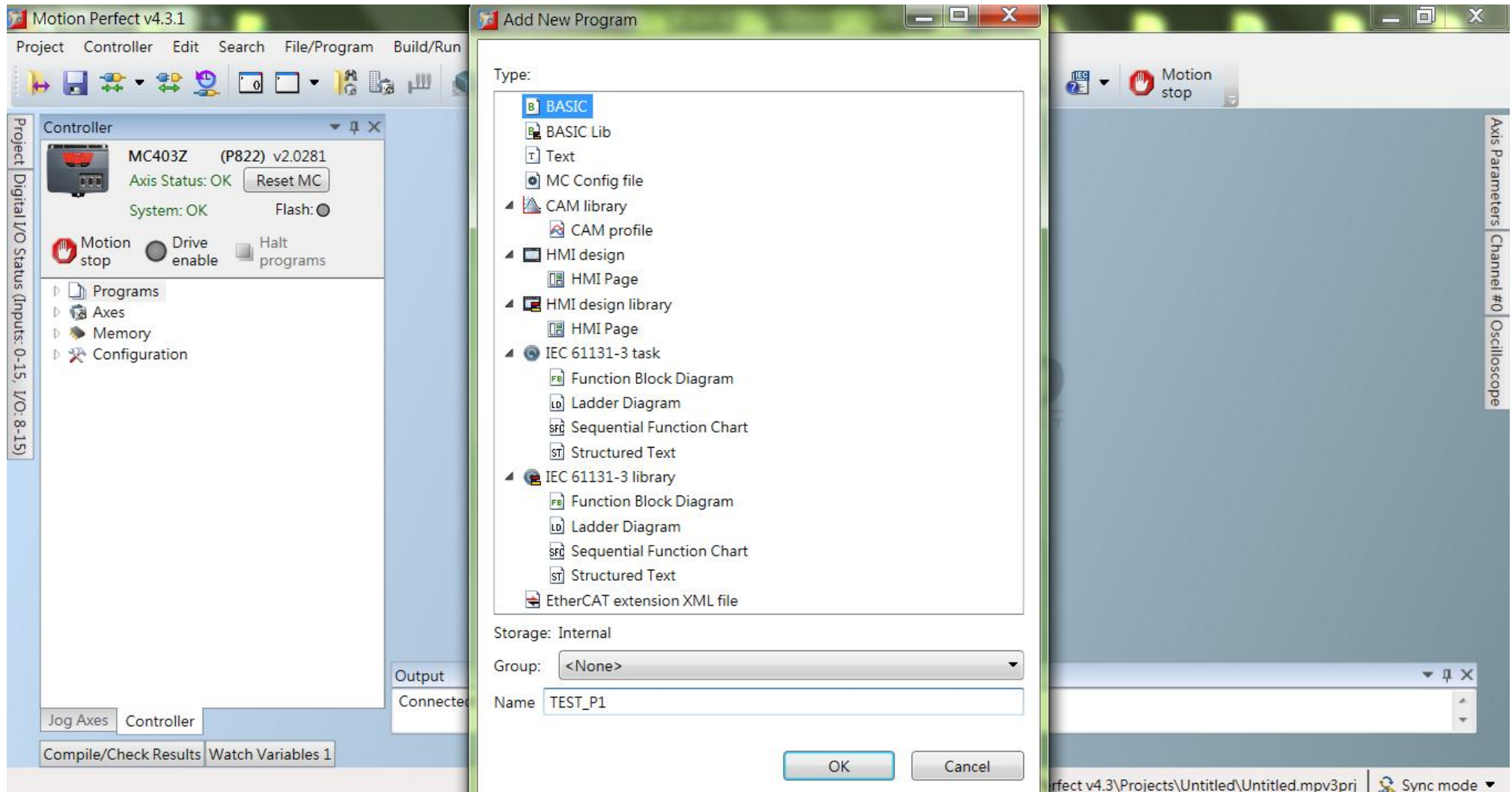
- Axis Parameters:** A table showing parameters for Axis (0) and Axis (1). The UNITS parameter is highlighted in yellow, with a value of 400000.0 for Axis (0) and 1.0 for Axis (1). Other parameters include Gains, Velocity profile, and Limits.
- Terminal:** A window showing the execution of commands. The output includes: WDOG=ON, >>BASE (0), >>UNITS=25000*16, >>ACCEL=10, >>DECEL=10, >>SPEED=1, >>MOVE (1), and >>. A green cursor is visible at the end of the last line.
- Bottom Status Bar:** Displays system information including MC403Z, Ethernet,192.168.0.250, C:\Users\Justin\Documents\Motion Perfect v4.3\Projects\Untitled\Untitled.mpv3prj, and Sync mode.

Parameter	Axis (0)	Axis (1)
ATYPE	Step+Dir	Step+Dir
UNITS	400000.0	1.0
Gains		
P_GAIN	1.0	1.0
I_GAIN	0.0	0.0
D_GAIN	0.0	0.0
OV_GAIN	0.0	0.0
VFF_GAIN	0.0	0.0
Velocity profile		
ACCEL	10.0	10000.0
DECEL	10.0	10000.0
SPEED	1.0	1000.0
CREEP	0.00025	100.0
MERGE	0	0
SRAMP	0	0
MSPEED	0.0	0.0
VP_SPEED	0.0	0.0
Limits		
DATUM_IN	-1	-1
FE_LIMIT	0.05000	20000.0
FE_RANGE	0.02500	10000.0
FHOLD_IN	-1	-1
FWD_IN	-1	-1
REP_DIST	500000.0	200000000000.0
REP_OPTION	0	0
REV_IN	-1	-1

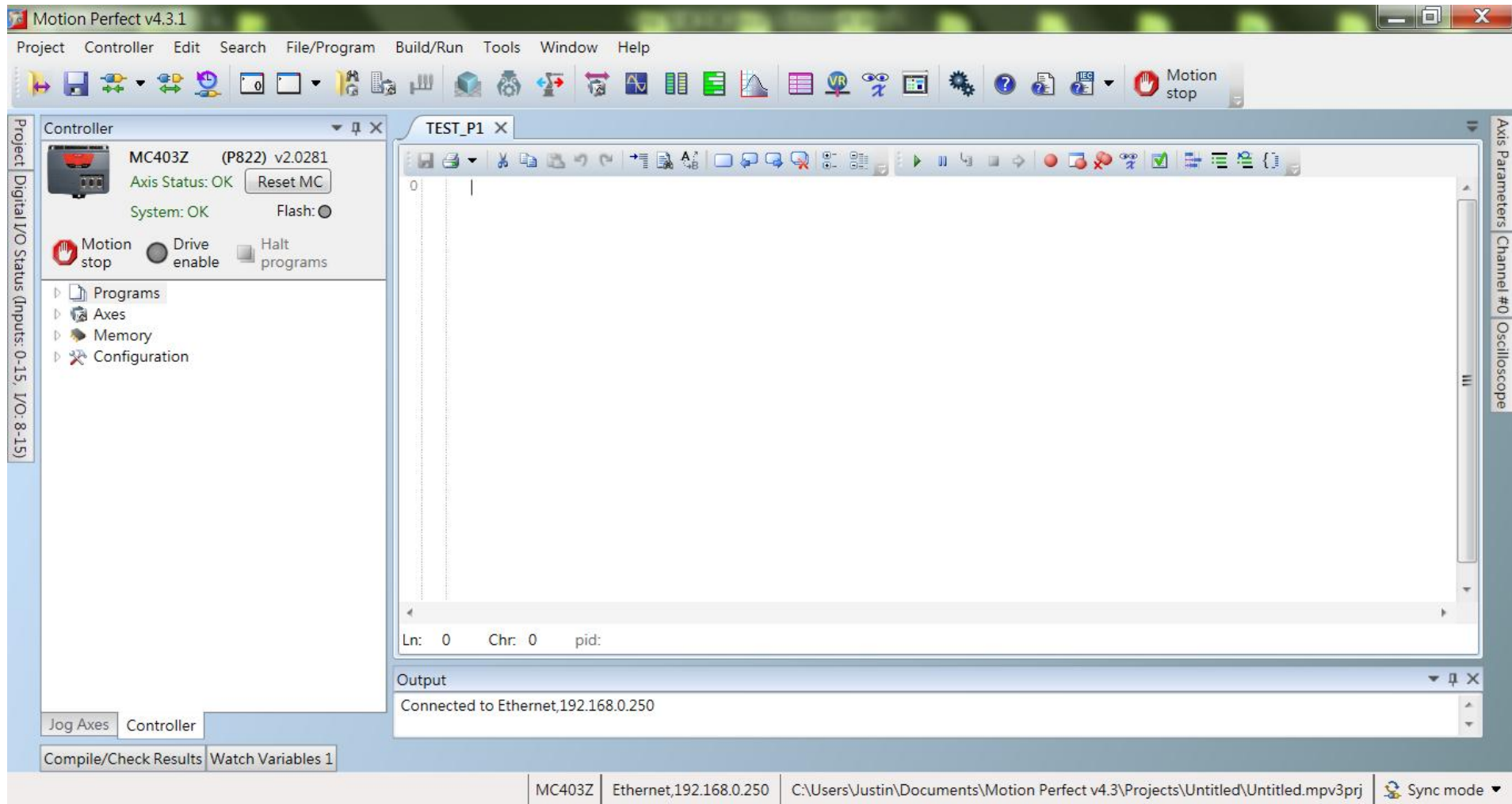
於 Controller 視窗的 Programs 點擊右鍵，點選 New...後會出現 Add New Program 視窗。



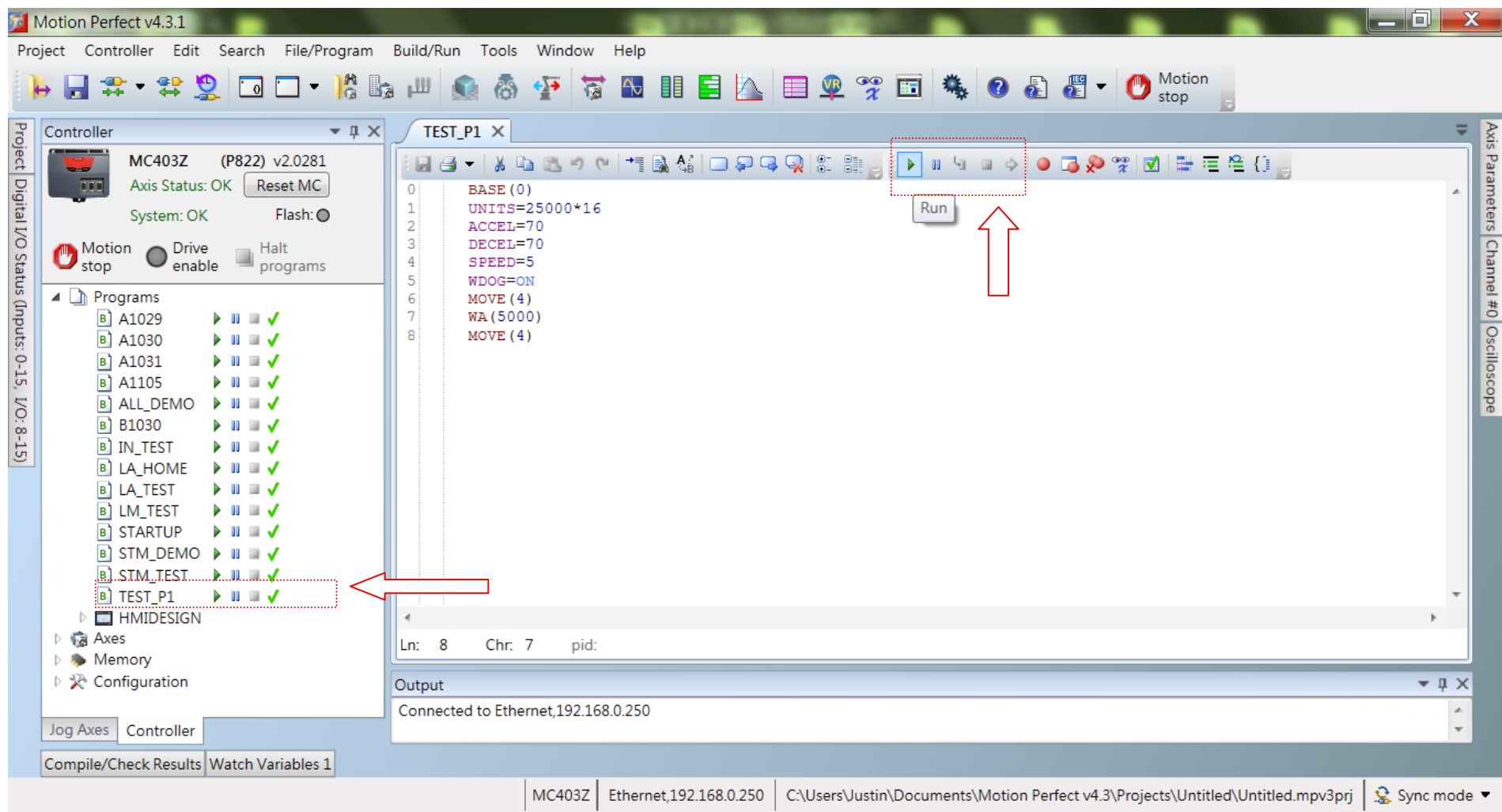
選擇 BASIC 並在 Name 欄位打上名字後，點擊 OK。




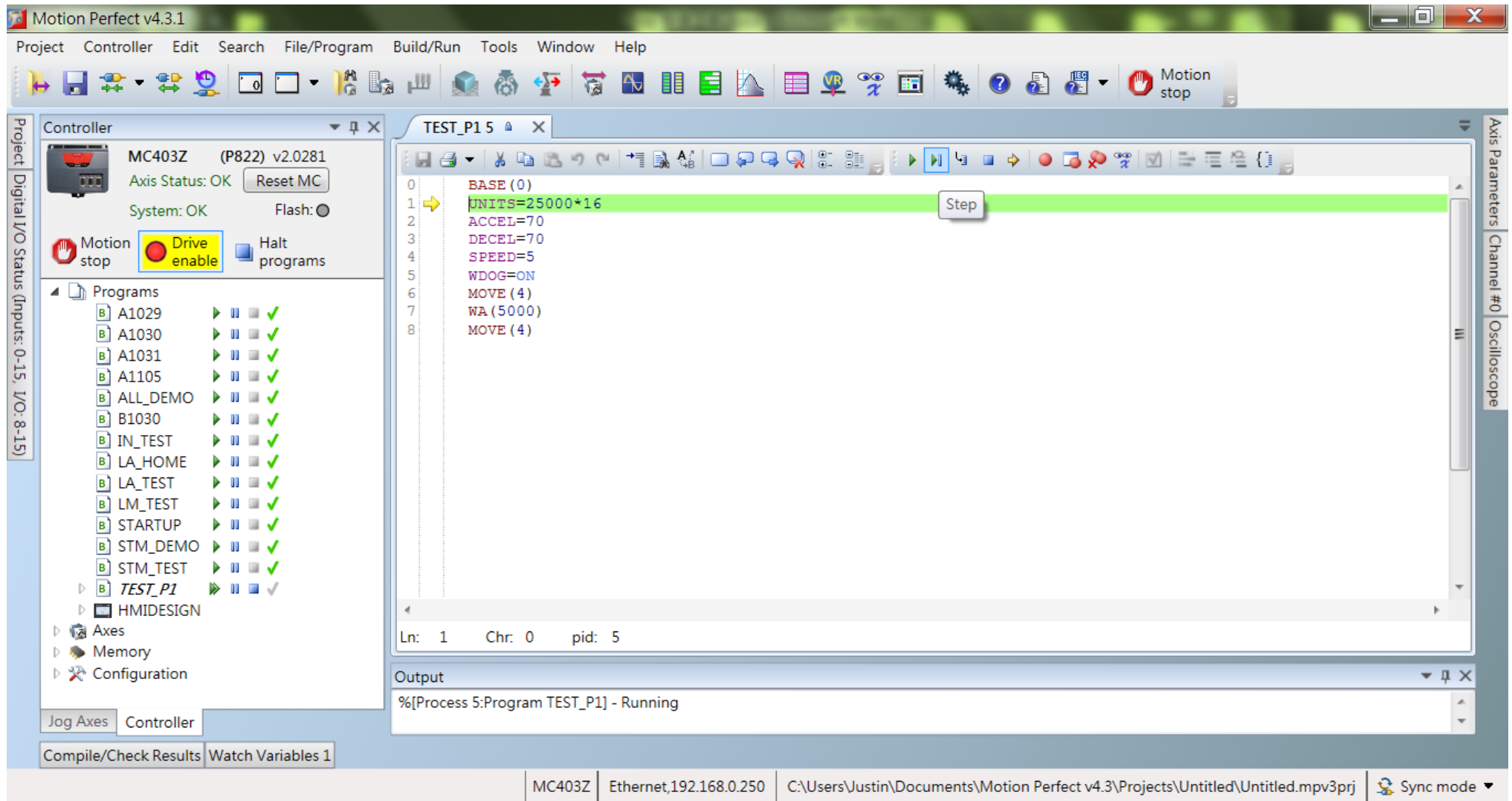
出現程式編輯介面，可開始編輯程式碼。





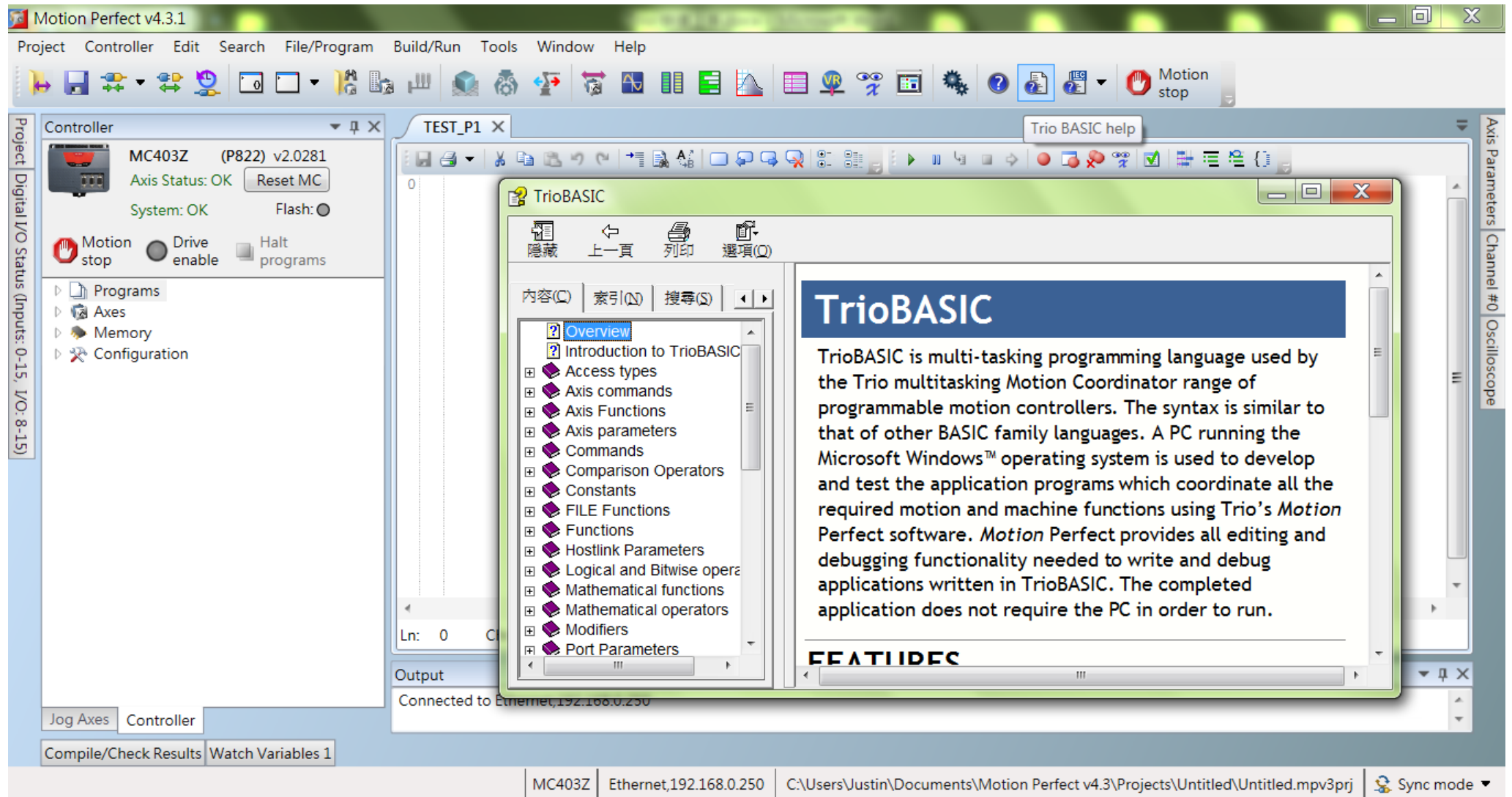
程式編輯完成後，按下 Run 鍵 ▶ 執行程式，程式由第一行開始執行至最後一行結束程式。
倘若程式尚未執行至最後一行，但需強制終止程式時，按下 Stop 鍵 ◻ 可結束程式。



按下 Pause/Step 鍵 ，一次只執行一行程式碼。



點擊圖示  程式指令幫助，可查詢指令。 點擊圖示 ，軟體操作說明。



The screenshot displays the Motion Perfect v4.3.1 software interface. The main window is titled "TEST_P1" and contains a code editor. A "TrioBASIC help" window is open, showing the "Overview" section. The help window includes a table of contents on the left and a main text area on the right. The table of contents lists various topics such as "Introduction to TrioBASIC", "Access types", "Axis commands", "Axis Functions", "Axis Parameters", "Commands", "Comparison Operators", "Constants", "FILE Functions", "Functions", "Hostlink Parameters", "Logical and Bitwise operations", "Mathematical functions", "Mathematical operators", "Modifiers", and "Port Parameters". The main text area provides a detailed overview of the TrioBASIC programming language, stating that it is a multi-tasking language used by the Trio multitasking Motion Coordinator range of programmable motion controllers. It notes that the syntax is similar to other BASIC family languages and that the software is used to develop and test application programs that coordinate motion and machine functions. The help window also features a "FEATURES" section at the bottom.

Motion Perfect v4.3.1

Project Controller Edit Search File/Program Build/Run Tools Window Help

Controller

MC403Z (P822) v2.0281

Axis Status: OK Reset MC

System: OK Flash:

Motion stop Drive enable Halt programs

Programs

Axes

Memory

Configuration

Project Digital I/O Status (Inputs: 0-15, I/O: 8-15)

TEST_P1 X Trio BASIC help

TrioBASIC

內容(C) 索引(I) 搜尋(S)

- Overview
- Introduction to TrioBASIC
- Access types
- Axis commands
- Axis Functions
- Axis Parameters
- Commands
- Comparison Operators
- Constants
- FILE Functions
- Functions
- Hostlink Parameters
- Logical and Bitwise operations
- Mathematical functions
- Mathematical operators
- Modifiers
- Port Parameters

TrioBASIC

TrioBASIC is multi-tasking programming language used by the Trio multitasking Motion Coordinator range of programmable motion controllers. The syntax is similar to that of other BASIC family languages. A PC running the Microsoft Windows™ operating system is used to develop and test the application programs which coordinate all the required motion and machine functions using Trio's *Motion Perfect* software. *Motion Perfect* provides all editing and debugging functionality needed to write and debug applications written in TrioBASIC. The completed application does not require the PC in order to run.

FEATURES

Ln: 0 C

Output

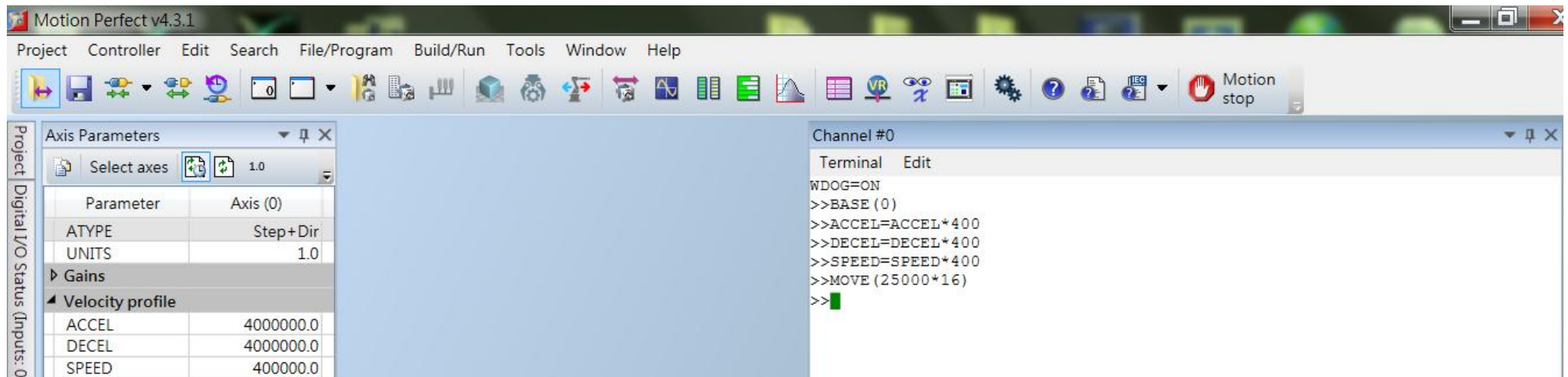
Connected to Ethernet,192.168.0.250

Jog Axes Controller

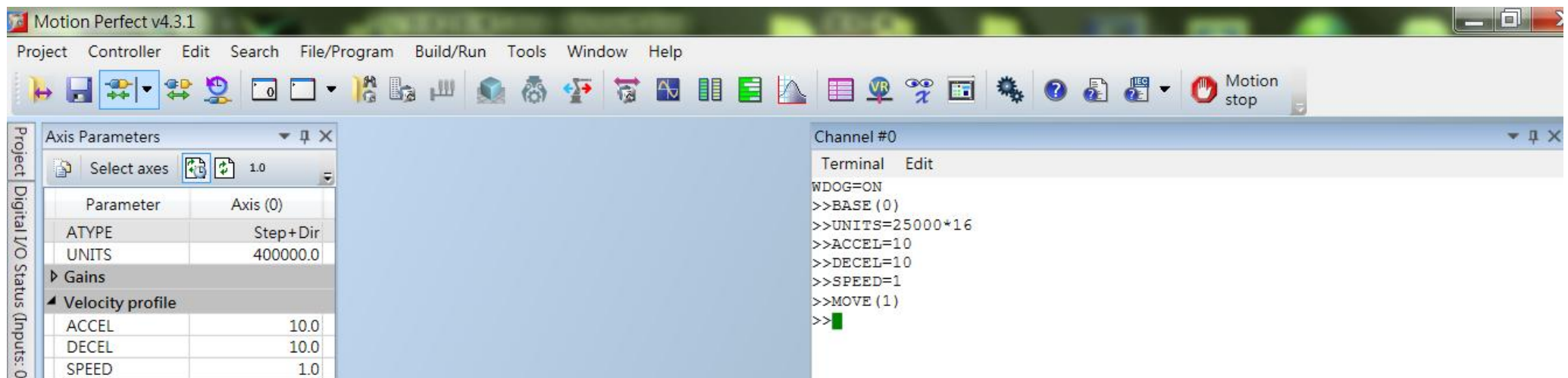
Compile/Check Results Watch Variables 1

MC403Z Ethernet,192.168.0.250 C:\Users\Justin\Documents\Motion Perfect v4.3\Projects\Untitled\Untitled.mpv3prj Sync mode

以步進馬達解析度 25000 為例，若 UNITS=1，其餘值乘上 $1/0.0025=400$ 倍，下指令 MOVE(25000*16)可使馬達 1 秒內轉 1 圈



以步進馬達解析度 25000 為例，UNITS=25000*16; ACCEL=10; DECEL=10; SPEED=1，下指令 MOVE(1)可使馬達 1 秒內轉 1 圈



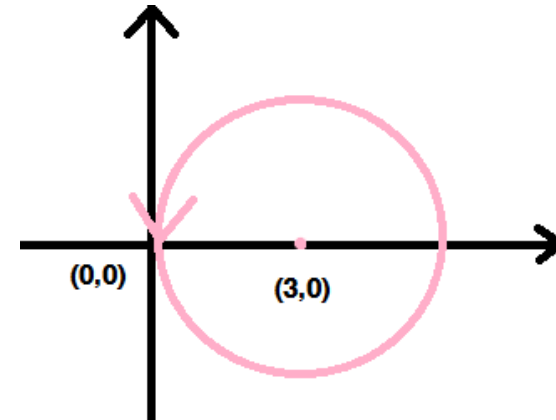
BASE(0)	'指定單運動軸位址為 0(可隨意指定)
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=90	'設定軸 0 加速度(UNITS/SEC/SEC)為 90 rev/s ²
DECEL=90	'設定軸 0 減速度(UNITS/SEC/SEC)為 90 rev/s ²
SPEED=1	'設定軸 0 速度(UNITS/SEC)為 1 rps
BASE(2)	'指定單運動軸位址為 2(可隨意指定)
UNITS=25000*16	'設定軸 2 馬達轉一圈步數
ACCEL=100	'設定軸 2 加速度(UNITS/SEC/SEC)為 100 rev/s ²
DECEL=100	'設定軸 2 減速度(UNITS/SEC/SEC)為 100 rev/s ²
SPEED=4	'設定軸 2 速度(UNITS/SEC)為 4 rps
WDOG=ON	'啟動運動
MOVE (5) AXIS(0)	'軸 0 前進五圈(軸 0 速度)
MOVE (5) AXIS(2)	'軸 2 前進五圈(軸 2 速度)
WAIT IDLE AXIS(0)	'等待軸 0 運動完成
WAIT IDLE AXIS(2)	'等待軸 2 運動完成

BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=90	'設定軸 0 加速度(UNITS/SEC/SEC)為 90 rev/s ²
DECEL=90	'設定軸 0 減速度(UNITS/SEC/SEC) 為 90 rev/s ²
SPEED=1	'設定速度(UNITS/SEC)為 1 rps (多軸補間運動之主速度)
BASE(2)	'指定單運動軸位址為 2
UNITS=25000*16	'設定軸 2 馬達轉一圈步數
ACCEL=100	'設定軸 2 加速度(UNITS/SEC/SEC)為 100 rev/s ²
DECEL=100	'設定軸 2 減速度(UNITS/SEC/SEC)為 100 rev/s ²
SPEED=4	'設定軸 2 速度(UNITS/SEC)為 4 rps
WDOG=ON	'啟動運動
BASE(0,2)	'指定多運動軸位址為 0 與 2
MOVE(1,5)	'軸 0(配合主速度自動調整)前進一圈;軸 2(圈數多的行主速度)前進五圈
WAIT IDLE	'等待多軸運動完成

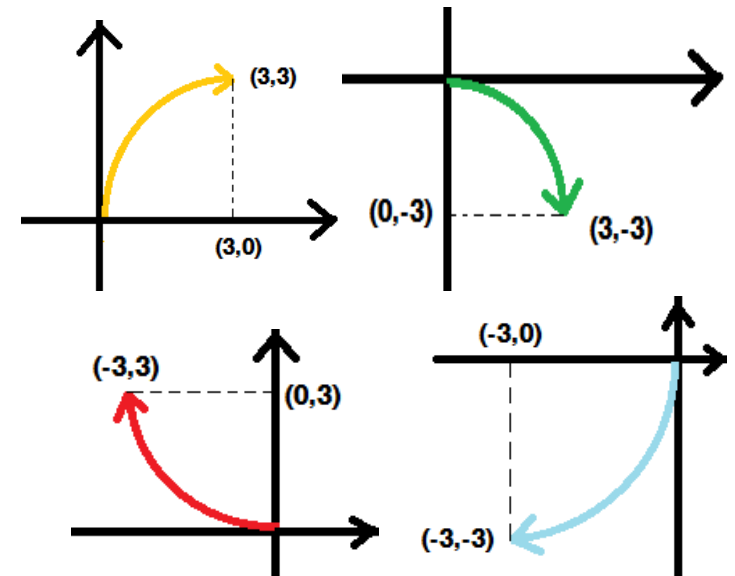
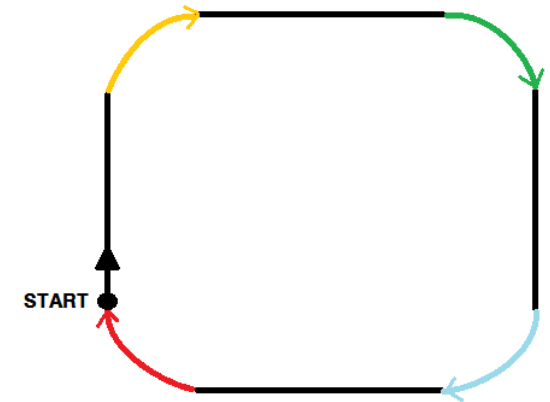
BASE(0) '指定單運動軸位址為 0
 UNITS=25000*16 '設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
 ACCEL=100 '設定軸 0 加速度(UNITS/SEC/SEC)為 100 rev/s²
 DECEL=100 '設定軸 0 減速度(UNITS/SEC/SEC)為 100 rev/s²
 SPEED=1 '設定軸 0 速度(UNITS/SEC)為 1 rps(主速度)

BASE(2) '指定單運動軸位址為 2
 UNITS=25000*16 '設定軸 2 馬達轉一圈步數
 ACCEL=100 '設定軸 2 加速度(UNITS/SEC/SEC)為 100 rev/s²
 DECEL=100 '設定軸 2 減速度(UNITS/SEC/SEC)為 100 rev/s²
 SPEED=5 '設定軸 2 速度(UNITS/SEC)為 5 rps

'draw a circle
 WDOG=ON '啟動運動
 BASE(0,2) '指定多運動軸位址為 0 與 2
 MOVECIRC(0,0,3,0,0) '圓弧以主速度相對移動(終點(0,0);圓心(3,0);方向:逆時針)
 WAIT IDLE '等待多軸運動完成

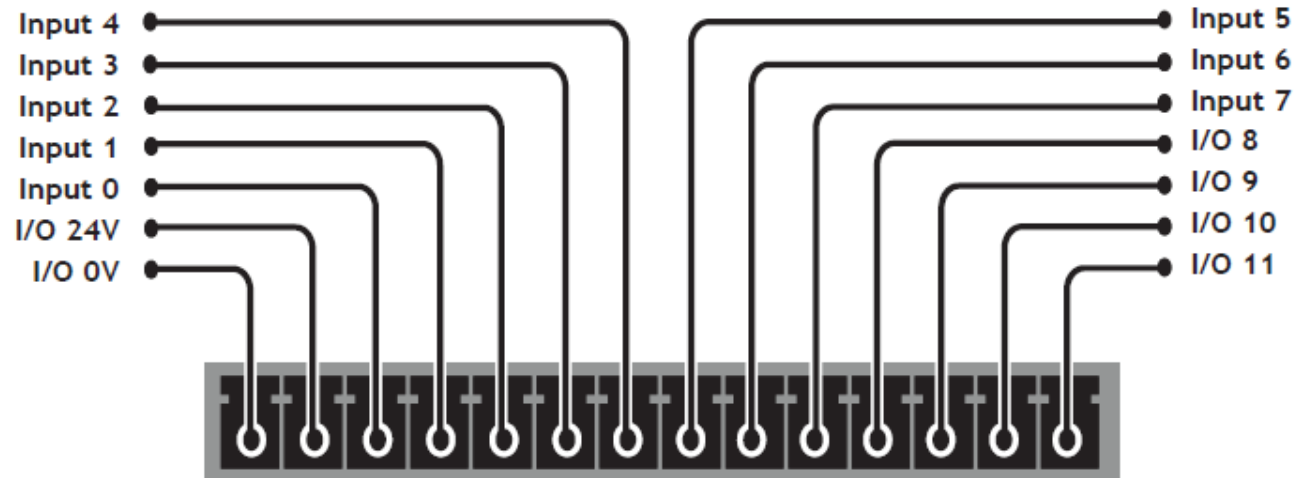
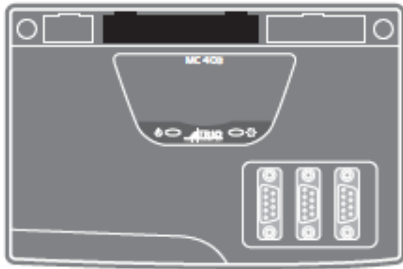


BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=100	'設定軸 0 加速度(UNITS/SEC/SEC)為 100 rev/s ²
DECEL=100	'設定軸 0 減速度(UNITS/SEC/SEC)為 100 rev/s ²
SPEED=1	'設定軸 0 速度(UNITS/SEC)為 1 rps(主速度)
BASE(2)	'指定單運動軸位址為 2
UNITS=25000*16	'設定軸 2 馬達轉一圈步數
ACCEL=100	'設定軸 2 加速度(UNITS/SEC/SEC)為 100 rev/s ²
DECEL=100	'設定軸 2 減速度(UNITS/SEC/SEC)為 100 rev/s ²
SPEED=5	'設定軸 2 速度(UNITS/SEC)為 5 rps
WDOG=ON	'啟動運動
BASE(0,2)	'指定多運動軸位址為 0 與 2
MERGE=ON	'啟動融合
MOVE(0,6)	'軸 0(配合主速度自動調整)前進 0 圈;軸 2(圈數多的行主速度)前進 6 圈
MOVECIRC(3,3,3,0,1)	'圓弧以主速度相對移動(終點(3,3);圓心(3,0);方向:順時針) (橙色)
MOVE(6,0)	'軸 0(圈數多的行主速度)前進 6 圈;軸 2(配合主速度自動調整)前進 0 圈
MOVECIRC(3,-3,0,-3,1)	'圓弧以主速度相對移動(終點(3,-3);圓心(0,-3);方向:順時針) (綠色)
MOVE(0,-6)	'軸 0(配合主速度自動調整)前進 0 圈;軸 2(圈數多的行主速度)後退 6 圈
MOVECIRC(-3,-3,-3,0,1)	'圓弧以主速度相對移動(終點(-3,-3);圓心(-3,0);方向:順時針) (藍色)
MOVE(-6,0)	'軸 0(圈數多的行主速度)後退 6 圈;軸 2(配合主速度自動調整)前進 0 圈
MOVECIRC(-3,3,0,3,1)	'圓弧以主速度相對移動(終點(-3,3);圓心(0,3);方向:順時針) (紅色)
WAIT IDLE	'等待運動完成
MERGE=OFF	'關閉融合

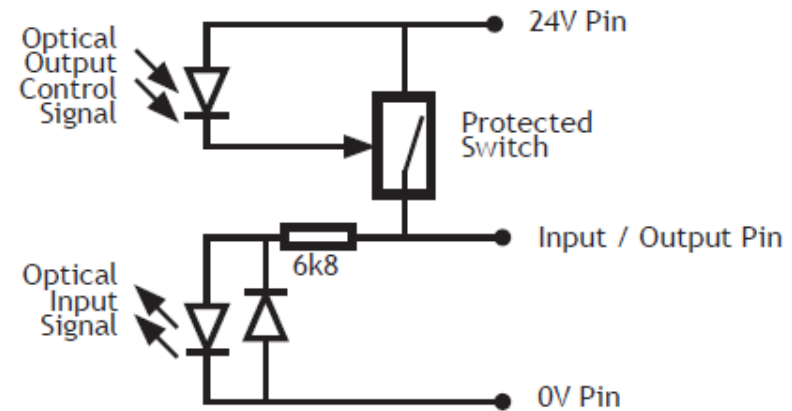
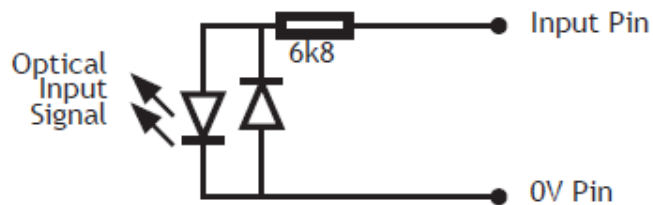




I/O CONNECTOR 1

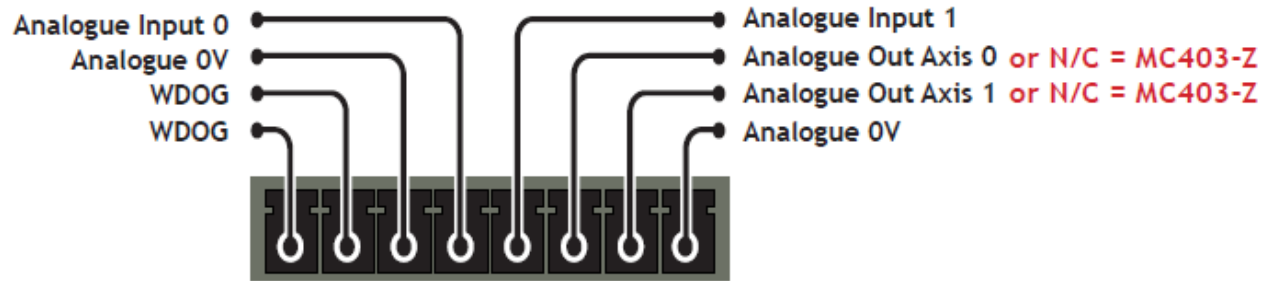
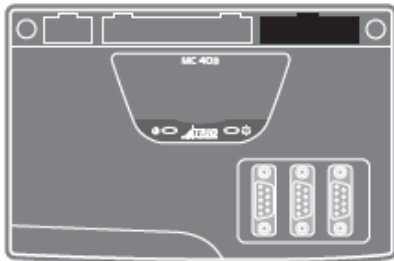


24V Power / Inputs / I/O



Any input, 0 - 5, may be mapped to any axis as a registration input.

I/O CONNECTOR 2

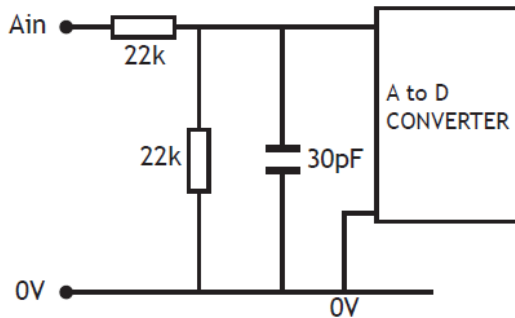


WDOG / Analogue Inputs / Outputs

I/O 24V must be applied to power the voltage outputs. (MC403 ONLY)

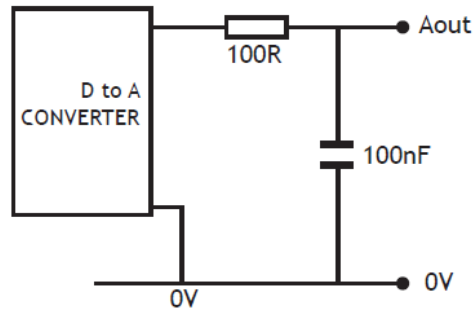
ANALOGUE INPUTS

AIN0: 0 TO 10V
 AIN1: 0 TO 10V



ANALOGUE OUTPUTS

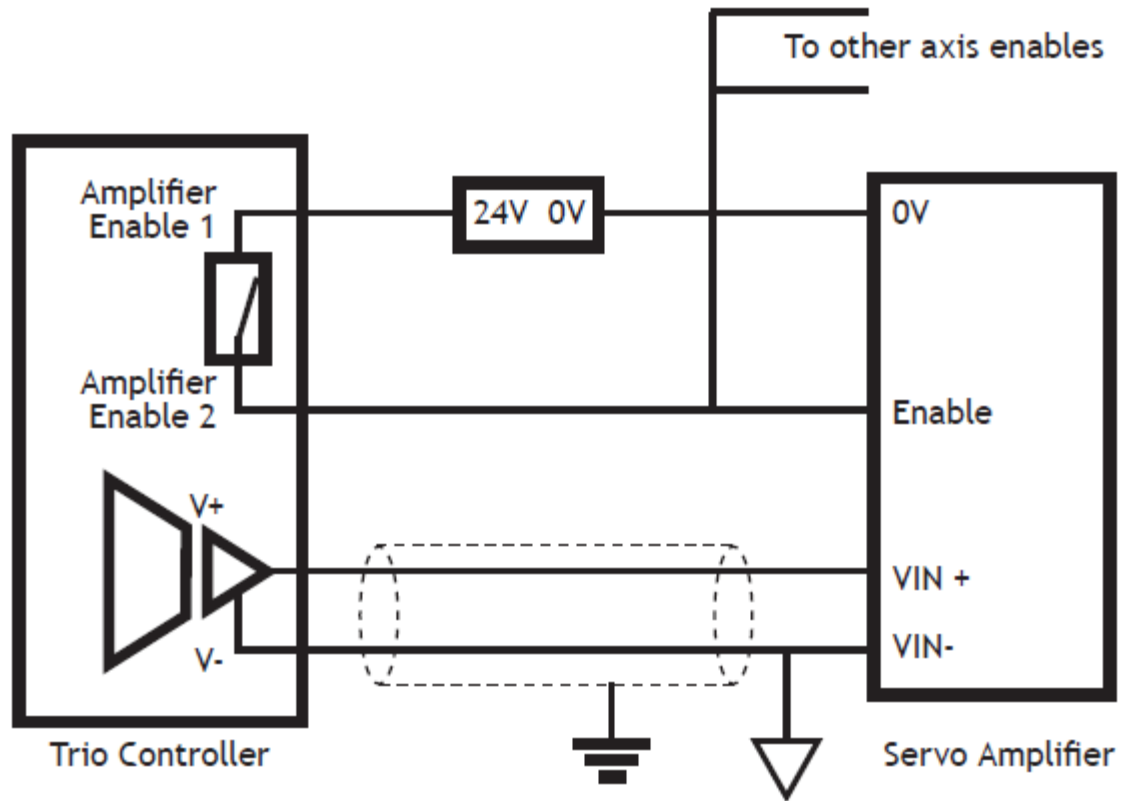
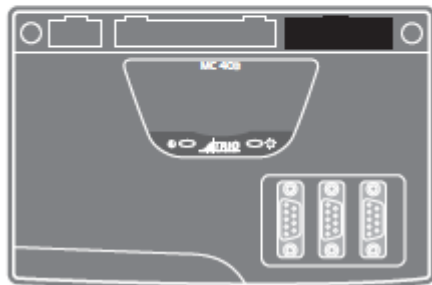
AOUT 0 TO AOUT 1
 Output: +/-10V at 5mA
 Output impedance: 100 Ohms.
 Common 0V return. Isolated from I/O & Encoders.



MC403 ONLY

ALL STEPPER AND SERVO AMPLIFIERS MUST BE INHIBITED WHEN THE AMPLIFIER ENABLE OUTPUT IS OPEN CIRCUIT

**AMPLIFIER
ENABLE
(WATCHDOG)
RELAY OUTPUT**



BASE(0)	'指定單運動軸位址為 0	b=0	'設定參數 b 為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)	WHILE b<1	'迴圈開始(當 b 小於 1 條件成立時)
ACCEL=70	'設定軸 0 加速度(UNITS/SEC/SEC)為 70 rev/s^2	IF IN(0)=ON THEN	'假如輸入(0)為 HIGH，則
DECEL=70	'設定軸 0 減速度(UNITS/SEC/SEC)為 70 rev/s^2	MOVEABS(0)	'移動到位置 0
SPEED=5	'設定軸 0 速度(UNITS/SEC)為 5 rps	WAIT IDLE	'軸 0 等待運動完成
WDOG=ON	'啟動運動	b=b+1	'參數 b 加 1
DEFPOS(0)	'設定當前位置為 0	ENDIF	'結束假設句
WHILE TRUE	'迴圈開始(無窮迴圈)	WEND	'迴圈結束
c=0	'設定參數 c 為 0	WEND	'迴圈結束
WHILE c<5	'迴圈開始(當 c 小於 5 條件成立時)		
IF IN(0)=ON THEN	'假如輸入(0)為 HIGH，則		
MOVE (1) AXIS(0)	'軸 0 前進 1 圈		
WAIT IDLE AXIS(0)	'軸 0 等待運動完成		
OP(8,ON)	'使輸出(8)為 HIGH,即燈亮		
WA(500)	'等待 0.5 sec		
OP(8,OFF)	'使輸出(8)為 LOW,即燈滅		
WA(500)	'等待 0.5 sec		
c=c+1	'參數 c 加 1		
ENDIF	'結束假設句		
WEND	'迴圈結束		

BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=70	'設定軸 0 加速度(UNITS/SEC/SEC)為 70 rev/s^2
DECEL=70	'設定軸 0 減速度(UNITS/SEC/SEC)為 70 rev/s^2
SPEED=5	'設定軸 0 速度(UNITS/SEC)為 5 rps
WDOG=ON	'啟動運動
REPEAT	'迴圈開始
MOVE (1) AXIS(0)	'軸 0 前進 1 圈
OP(8,ON)	'使輸出(8)為 HIGH,即燈亮
WA(500)	'等待 0.5 sec
OP(8,OFF)	'使輸出(8)為 LOW,即燈滅
WA(500)	'等待 0.5 sec
UNTIL IN(0) = 1	'迴圈結束(當輸入(0)為 HIGH 時)

BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=10	'設定軸 0 加速度(UNITS/SEC/SEC)為 10 rev/s^2
DECEL=10	'設定軸 0 減速度(UNITS/SEC/SEC)為 10 rev/s^2
SPEED=1	'設定軸 0 速度(UNITS/SEC)為 1 rps
WDOG=ON	'啟動運動
WHILE TRUE	'迴圈開始(無窮迴圈)
IF IN(0)=ON THEN	'假如輸入(0)為 HIGH，則
FORWARD	'以 SPEED 速度持續前進運動(正轉)
WAIT UNTIL IN(0)=OFF	'在此等待，直到輸入(0)=LOW 成立時離開
CANCEL	'取消運動
WAIT IDLE	'等待運動完成
ENDIF	'結束假設句
IF IN(1)=ON THEN	'假如輸入(1)為 HIGH，則
REVERSE	'以 SPEED 速度持續後退運動(反轉)
WAIT UNTIL IN(1)=OFF	'在此等待，直到輸入(1)=LOW 成立時離開
CANCEL	'取消運動
WAIT IDLE	'等待運動完成
ENDIF	'結束假設句
WEND	'迴圈結束

BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=10	'設定軸 0 加速度(UNITS/SEC/SEC)為 10 rev/s^2
DECEL=10	'設定軸 0 減速度(UNITS/SEC/SEC)為 10 rev/s^2
SPEED=1	'設定軸 0 速度(UNITS/SEC)為 1 rps
WDOG=ON	'啟動運動
WHILE TRUE	'迴圈開始(無窮迴圈)
INVERT_IN(0,ON)	'使 input(0)狀態為 HIGH
INVERT_IN(1,ON)	'使 input(1)狀態為 HIGH
FWD_JOG=0	'當 input(0)=OFF 時,馬達以 JOGSPEED 速度持續正轉; 當 input(0)=ON 時,馬達停止
REV_JOG=1	'當 input(1)=OFF 時,馬達以 JOGSPEED 速度持續反轉; 當 input(1)=ON 時,馬達停止
JOGSPEED = 4	'設定軸 0 馬達正反轉時速度(UNITS/SEC)為 4 rps
WEND	'迴圈結束

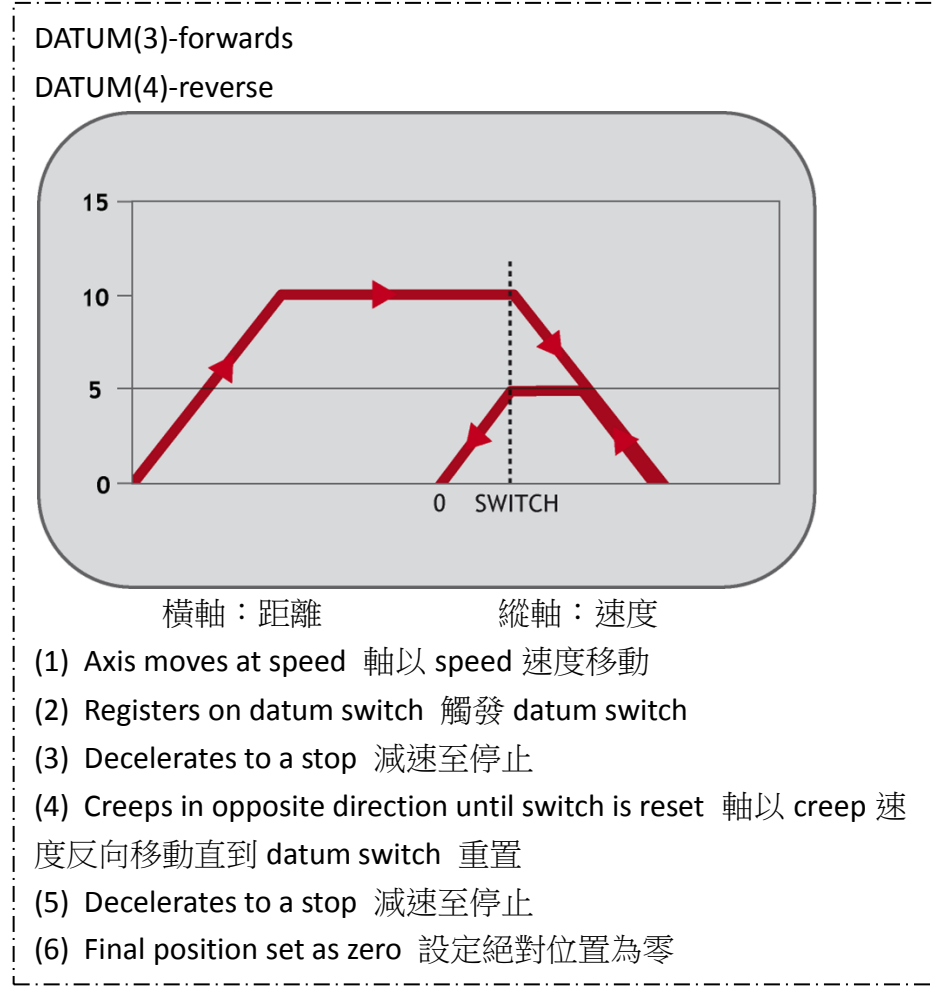
'Program "IN_TEST"

WHILE TRUE	'迴圈開始(無窮迴圈)
IF (IN(0)=1 AND PROC_STATUS PROC(1)=0) THEN	'假如輸入(0)為 HIGH 且程式序列(1)為未執行狀態，則
RUN "TEST_P1",1	'執行程式"TEST_P1"，標註序列(1)
ENDIF	'結束假設句
IF (IN(1)=1) THEN	'假如輸入(1)為 HIGH，則
STOP "TEST_P1"	'停止程式"TEST_P1"
ENDIF	'結束假設句
WEND	'迴圈結束

'Program "TEST_P1"

BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=70	'設定軸 0 加速度(UNITS/SEC/SEC)為 70 rev/s ²
DECEL=70	'設定軸 0 減速度(UNITS/SEC/SEC)為 70 rev/s ²
SPEED=5	'設定軸 0 速度(UNITS/SEC)為 5 rps
WDOG=ON	'啟動運動
MOVE(4)	'軸 0 前進四圈
WAIT IDLE	'等待運動完成

WDOG=ON	'啟動運動
BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
ACCEL=100	'設定軸 0 加速度(UNITS/SEC/SEC)為 100 rev/s ²
DECEL=100	'設定軸 0 減速度(UNITS/SEC/SEC)為 100 rev/s ²
SPEED=10	'設定軸 0 速度(UNITS/SEC)為 10 rps
CREEP=5	'設定軸 0 爬行速度(UNITS/SEC)為 5 rps (DATUM 時使用)
DATUM_IN=0	'設定輸入(0)為 datum switch, 當 LOW 時觸發
INVERT_IN(0,1)	'設定輸入(0)=HIGH
'home	
DATUM(4)	'選擇 DATUM(4):反向運動尋找 datum switch 並設定原點
WAIT IDLE	'等待運動完成
INVERT_IN(0,0)	'設定輸入(0)=LOW



BASE(0)	'指定單運動軸位址為 0
UNITS=25000*16	'設定軸 0 馬達轉一圈步數(編碼器解析度 25000;步進馬達需乘 16 倍數)
WDOG=ON	'啟動運動
WHILE 1	'迴圈開始(無窮迴圈)
GOSUB aaa	'前往附程式 aaa
GOSUB bbb	'前往附程式 bbb
WEND	'迴圈結束
aaa:	'附程式 aaa 開始
ACCEL=10	'設定軸 0 加速度(UNITS/SEC/SEC)為 10 rev/s ²
DECEL=10	'設定軸 0 減速度(UNITS/SEC/SEC)為 10 rev/s ²
SPEED=1	'設定軸 0 速度(UNITS/SEC)為 1 rps
MOVE(3)	'軸 0 前進 3 圈
WAIT IDLE	'等待運動完成
WA(2000)	'等待 2 sec
RETURN	'附程式 aaa 結束
bbb:	'附程式 bbb 開始
ACCEL=50	'設定軸 0 加速度(UNITS/SEC/SEC)為 50 rev/s ²
DECEL=50	'設定軸 0 減速度(UNITS/SEC/SEC)為 50 rev/s ²
SPEED=5	'設定軸 0 速度(UNITS/SEC)為 5 rps
MOVE(10)	'軸 0 前進 10 圈
WAIT IDLE	'等待運動完成
WA(2000)	'等待 2 sec
RETURN	'附程式 bbb 結束

指令說明 (此處只列出練習程式中所用到指令，其餘指令需查閱手冊)

[目錄](#)

AXIS(expression) Assigns ONE command, function or axis parameter operation to a particular axis.

ACCEL The ACCEL axis parameter may be used to set or read back the acceleration rate of each axis fitted. The acceleration rate is in UNITS/sec/sec.

ACC(rate) Sets both the acceleration and deceleration rate simultaneously.

<expression1> AND <expression2> This performs an AND function between corresponding bits of the integer part of two valid TrioBASIC expressions. The

AND function between two bits is defined as follows:

AND	0	1
0	0	0
1	0	1

BASE(axis no<,second axis><,third axis>...) The BASE command is used to direct all subsequent motion commands and axis parameter read/writes to a particular axis, or group of axes.

CANCEL([mode]) Used to cancel current or buffered axis commands on an axis or an interpolating axis group.

CREEP The creep speed is used for the slow part of a DATUM sequence.

DATUM Performs one of 6 datuming sequences to locate an axis to an absolute position.

DATUM_IN This parameter holds a digital input channel to be used as a datum input.

DECEL The DECEL axis parameter may be used to set or read back the deceleration rate of each axis fitted.

DEFPOS(pos1 [,pos2[, pos3[, pos4...]]]) Defines the current position(s) as a new absolute value.

FORWARD Sets continuous forward movement.

FWD_JOG=value This parameter holds the input number to be used as a jog forward input. When the FWD_JOG input is active the axis moves in forward at JOGSPEED. Value=-1: Disable the input as FWD_JOG (default); Value=0~63: Input to use as datum input.

GOSUB..RETURN Stores the position of the line after the GOSUB command and then branches to the label specified. Upon reaching the RETURN statement, control is returned to the stored line.

IN([input_no[,final_input]]) IN is used to read the state of the inputs.

INVERT_IN(input, state) The INVERT_IN command allows the input channels to be individually inverted in software. State=ON: the input is inverted in software; State=OFF :the input is not inverted.

IF..THEN..ELSEIF..ELSE..ENDIF An IF program structure is used to execute a block of code after a valid expression. If no expressions are valid and an ELSE is

present the commands under the ELSE will be executed.

JOGSPEED Sets the jog speed in user units for an axis to run at when performing a jog.

MOVE(distance1 [,distance2 [,distance3 [,distance4...]]) Incremental move.

MOVEABS(position1[, position2[, position3[, position4...]]) Absolute position move.

MOVECIRC(end1, end2, centre1, centre2, direction [,ta [,output]]) Moves 2 orthogonal axes in such a way as to produce a circular arc at the tool point.

MERGE Velocity profiled moves can be MERGEed together so that the speed will not ramp down to zero between the current move and the buffered move.

OP(output, state) Set the state of an individual output.

PROC Allows a particular process to be specified when using a Process Parameter, Function or Command.

PROC_STATUS Returns the status of another process, referenced with the PROC (x) modifier.

REVERSE Sets continuous reverse movement.

REPEAT .. UNTIL The REPEAT..UNTIL construct allows a block of commands to be continuously repeated until an expression becomes TRUE .

REV_JOG=value This parameter holds the input number to be used as a jog reverse input. When the REV_JOG input is active the axis moves in reverse at JOGSPEED. Value=-1: Disable the input as REV_JOG (default); Value=0~63: Input to use as datum input.

RUN “program” [, process] Runs a named program on the controller.

RAPIDSTOP [(mode)] The RAPIDSTOP command cancels the currently executing move on ALL axes.

STOP “programe”,[process_number] Stops one program at its current line.

SPEED The SPEED axis parameter can be used to set/read back the demand speed axis parameter. The speed is in UNITS/sec.

UNITS The motion commands to set speeds, acceleration and moves use the UNITS scalar to allow values to be entered in more convenient units.

WA(time) Wait time(milliseconds)

WDOG Controls the WDOG relay contact used for enabling external drives. The WDOG=ON command MUST be issued in a program prior to executing moves.

WAIT UNTIL expression Suspends program execution until the expression is TRUE.

WAIT IDLE Suspend program execution until the move has finished.

WHILE .. WEND The commands contained in the WHILE..WEND loop are continuously executed until the condition becomes FALSE .