

HIWIN® MIKROSYSTEM



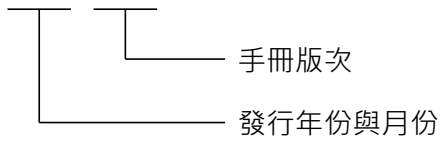
HIMC

Modbus TCP 使用手冊

修訂紀錄

手冊版次資訊亦標記於手冊封面右下角。

MH02UC01-2502_V1.0



發行日期	版次	適用軟體	更新內容
2025/02/28	1.0	iA Studio 3.1	<ol style="list-style-type: none">更新 2.1 節通訊介面。更新 3.2 節功能碼。更新 3.4 節資料類型。更新第 4 章資料暫存空間定義。
2022/06/30	0.3	iA Studio 2.0	Controller Information 資料暫存空間新增系統日期與時間。
2020/09/16	0.2	iA Studio 1.3	單位系統變更：m-rad-s → mm-deg-ms
2018/04/25	0.1	iA Studio 1.0.2461.0	第一版發行。

相關文件

透過相關文件，使用者可快速了解此手冊的定位，以及各手冊、產品之間的關聯性。詳細內容請至本公司官網→下載中心→手冊總覽閱覽 (https://www.hiwinmikro.tw/Downloads/ManualOverview_TC.htm)。

目錄

1.	簡介.....	1-1
1.1	HIMC Modbus TCP 介紹.....	1-2
2.	HIMC Modbus TCP 通訊介面.....	2-1
2.1	通訊介面.....	2-2
3.	HIMC Modbus TCP 功能介紹.....	3-1
3.1	資料暫存空間.....	3-2
3.2	功能碼.....	3-2
3.3	異常回覆.....	3-3
3.4	資料類型.....	3-3
4.	資料暫存空間定義.....	4-1
4.1	Coils.....	4-2
4.1.1	Axis.....	4-3
4.1.2	System Call.....	4-4
4.1.3	HMPL Task.....	4-5
4.2	Discrete Inputs.....	4-6
4.3	Input Registers.....	4-7
4.3.1	Axis.....	4-8
4.3.2	Controller Information.....	4-10
4.3.3	HMPL Task.....	4-11
4.4	Holding Registers.....	4-12
4.4.1	Axis.....	4-13
4.4.2	GPIO.....	4-15
4.4.3	Slave GPIO.....	4-16
4.4.4	User Table.....	4-17
4.4.5	User-defined Parameters.....	4-18
4.4.6	無定義.....	4-18

1. 簡介

1. 簡介.....	1-1
1.1 HIMC Modbus TCP 介紹.....	1-2

1.1 HIMC Modbus TCP 介紹

HIWIN 運動控制器 (HIWIN Motion Controller, HIMC) · 以下簡稱為 HIMC · 可支援 Modbus TCP 通訊協定。使用者可使用 HMI (Human Machine Interface) 或 PC 透過 Modbus TCP 存取 HIMC · 對各軸參數、系統呼叫及控制器資訊等項目進行讀取和寫入。

2. HIMC Modbus TCP 通訊介面

2.	HIMC Modbus TCP 通訊介面.....	2-1
2.1	通訊介面.....	2-2

2.1 通訊介面

HIMC 提供 CN3 和 CN4 連接孔作為與 PC 或 HMI 通訊使用。

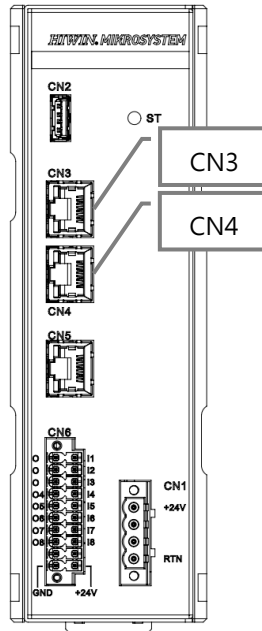


圖 2.1.1

CN3 和 CN4 連接孔的預設資訊如下：

表 2.1.1

連接孔	CN3	CN4
IP 位址	0.0.0.0	169.254.188.20
通訊埠	502	

註：

可透過 iA Studio 來設定 CN3 的 IP 位址，請參閱《iA Studio 軟體使用手冊》4.13 節 **IP Setting** 的說明。

HIMC 提供的虛擬控制器亦可由 Modbus TCP 存取。存取 HIMC 虛擬控制器的所需資訊如下：

表 2.1.2

虛擬控制器	
IP 位址	127.0.0.1
通訊埠	502

註：

使用 Modbus TCP 存取 HIMC 虛擬控制器前，請確保已連線至 HIMC 虛擬控制器。如需連線至 HIMC 虛擬控制器的詳細資訊，請參閱《iA Studio 軟體使用手冊》2.1.3 節**連線至虛擬控制器**。

3. HIMC Modbus TCP 功能介紹

3.	HIMC Modbus TCP 功能介紹.....	3-1
3.1	資料暫存空間.....	3-2
3.2	功能碼.....	3-2
3.3	異常回覆.....	3-3
3.4	資料類型.....	3-3

3.1 資料暫存空間

資料暫存空間主要分為四個區塊，HIMC 提供的資料暫存空間如下：

表 3.1.1

名稱	Coils / Register 位址	資料大小	存取類型
Coils	0X00000 ~ 0X65535	1 bit	讀 / 寫
Discrete Inputs	1X00000 ~ 1X65535	1 bit	唯讀
Input Registers	3X00000 ~ 3X65535	16 bit	唯讀
Holding Registers	4X00000 ~ 4X65535	16 bit	讀 / 寫

3.2 功能碼

HIMC 支援的功能碼 (function code) 如下：

表 3.2.1

功能碼	說明
01 (01 hex)	讀取 Coils 暫存空間。
02 (02 hex)	讀取 Discrete Inputs 暫存空間。
03 (03 hex)	讀取 Holding Registers 暫存空間。
04 (04 hex)	讀取 Input Registers 暫存空間。
05 (05 hex)	寫入單一 Coils 暫存空間。
06 (06 hex)	寫入單一 Holding Registers 暫存空間。
15 (0F hex)	寫入多個 Coils 暫存空間。
16 (10 hex)	寫入多個 Holding Registers 暫存空間。
23 (17 hex)	寫入並讀取多個 Holding Registers 暫存空間。

3.3 異常回覆

當接收到無法處理的要求時，HIMC 會回傳帶有異常碼的異常回覆。HIMC 的異常碼如下：

表 3.3.1

異常碼	定義	描述
01 (01 hex)	不合法的功能碼	接收到未支援的功能碼。 例：要求執行功能碼 20 (14 hex)。
02 (02 hex)	不合法的資料位址	存取未定義的暫存空間。 例：控制器提供 100 個 Register，要求從位址 96 開始，同時存取 5 個 Register，控制器便會回傳此異常碼。
03 (03 hex)	不合法的資料數值	存取的資料值不完整、存取的參數或 HMPL 全域變數不存在。 例：參數的起始位址為 0 並占用 2 個 Register。要求從位址 1 開始存取，或從位址 0 開始存取 1 個 Register，控制器便會回傳此異常碼。

3.4 資料類型

HIMC 內有不同資料類型的參數，HIMC 參數的資料類型如下：

表 3.4.1

資料類型	資料大小	數值範圍
int8_t	8 bit	-128 ~ 127
uint8_t	8 bit	0 ~ 255
int16_t	16 bit	-32,768 ~ 32,767
uint16_t	16 bit	0 ~ 65,535
int32	32 bit	-2,147,483,648 ~ 2,147,483,647
uint32	32 bit	0 ~ 4,294,967,295
int64	64 bit	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
uint64	64 bit	0 ~ 18,446,744,073,709,551,615
float	32 bit	3.4E +/- 38 (7 位數)

HIMC 會依照參數的資料類型，將參數資料存放到對應的暫存空間。使用者可依照以下說明讀取參數資料。

■ int8_t、uint8_t、int16_t 和 uint16_t

資料類型為 int8_t、uint8_t、int16_t 和 uint16_t 的參數，是用於數位輸入 / 輸出及顯示控制器狀態。參數資料的存放方式如下：

數值	Register N
17 (0x0011) (00000000 00010001)	0x0011 (00000000 00010001)

■ int32_t 和 uint32_t

資料類型為 int32_t 和 uint32_t 的參數，是用於數位輸入 / 輸出及顯示控制器狀態。參數資料的存放方式如下：

數值	Register N (開始位址)	Register N+1 (結束位址)
2097169 (0x00200011) (00000000 00100000 00000000 00010001)	0x0011 (00000000 00010001)	0x0020 (00000000 00100000)

■ float

資料類型為 float 的參數，參數資料的存放方式如下：

數值	Register N (開始位址)	Register N+1 (結束位址)
0.85 (0x3F59999A)	0x999A	0x3F59

■ double

資料類型為 double 的參數，參數資料的存放方式如下：

數值	Register N (開始位址)	Register N+1	Register N+2	Register N+3 (結束位址)
0.85 (0x3FEB333333333333)	0x3333	0x3333	0x3333	0x3FEB

4. 資料暫存空間定義

4.	資料暫存空間定義	4-1
4.1	Coils	4-2
4.1.1	Axis	4-3
4.1.2	System Call	4-4
4.1.3	HMPL Task.....	4-5
4.2	Discrete Inputs.....	4-6
4.3	Input Registers	4-7
4.3.1	Axis	4-8
4.3.2	Controller Information.....	4-10
4.3.3	HMPL Task.....	4-11
4.4	Holding Registers.....	4-12
4.4.1	Axis	4-13
4.4.2	GPIO	4-15
4.4.3	Slave GPIO	4-16
4.4.4	User Table.....	4-17
4.4.5	User-defined Parameters.....	4-18
4.4.6	無定義.....	4-18

4.1 Coils

Coils 暫存空間提供 HIMC 執行各軸的軸命令功能、系統呼叫與 HMPL Task。預設參數分類定義如表 4.1.1。最多可支援 128 軸的運動命令。

表.4.1.1

參數分類	描述
Axis	執行各軸的軸命令功能，如激磁、清除錯誤等。
System Call	執行系統呼叫，如緊急停止、吋動、相對移動等。
HMPL Task	執行或停止 HMPL Task。

4.1.1 Axis

使用者可存取 Axis 參數的暫存空間，執行點對點運動、激磁、清除錯誤與位置歸零的指令。

Axis 的資料暫存空間定義如下：

表.4.1.1.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明				
0 (0x0000)	選擇軸	bool	讀 / 寫	設定或顯示第 N 軸是否為選擇軸。 <table border="1" style="margin-left: 20px;"> <tr> <td style="background-color: #cccccc;">Bit 0</td> <td>0：取消選擇軸。</td> </tr> <tr> <td></td> <td>1：選擇軸。</td> </tr> </table> 註：執行系統呼叫時，只會對選擇軸進行運動控制。	Bit 0	0：取消選擇軸。		1：選擇軸。
Bit 0	0：取消選擇軸。							
	1：選擇軸。							
1 (0x0001)	P2P 重複點對點	bool	讀 / 寫	設定或顯示第 N 軸是否重複執行點對點運動。 <table border="1" style="margin-left: 20px;"> <tr> <td style="background-color: #cccccc;">Bit 0</td> <td>0：不重複執行。</td> </tr> <tr> <td></td> <td>1：重複執行。</td> </tr> </table> 註：由系統呼叫執行點對點運動。	Bit 0	0：不重複執行。		1：重複執行。
Bit 0	0：不重複執行。							
	1：重複執行。							
2 (0x0002)	激磁或解激磁	bool	讀 / 寫	設定或顯示第 N 軸為激磁或解激磁。 <table border="1" style="margin-left: 20px;"> <tr> <td style="background-color: #cccccc;">Bit 0</td> <td>0：解激磁。</td> </tr> <tr> <td></td> <td>1：激磁。</td> </tr> </table>	Bit 0	0：解激磁。		1：激磁。
Bit 0	0：解激磁。							
	1：激磁。							
3 (0x0003)	清除錯誤停止	bool	讀 / 寫	清除第 N 軸的錯誤狀態。 <table border="1" style="margin-left: 20px;"> <tr> <td style="background-color: #cccccc;">Bit 0</td> <td>1：清除錯誤停止。</td> </tr> </table>	Bit 0	1：清除錯誤停止。		
Bit 0	1：清除錯誤停止。							
4 (0x0004)	位置歸零	bool	讀 / 寫	將第 N 軸的位置歸零。 <table border="1" style="margin-left: 20px;"> <tr> <td style="background-color: #cccccc;">Bit 0</td> <td>1：位置歸零。</td> </tr> </table>	Bit 0	1：位置歸零。		
Bit 0	1：位置歸零。							

註：

*1：第 N 軸各個參數的位址 = Register 位址 + 16 * N (N_{max} = 127)

4.1.2 System Call

使用者可存取 System Call 參數的暫存空間，對各軸進行運動控制，如緊急停止、吋動、相對移動等。System Call 的資料暫存空間定義如下：

表.4.1.2.1

Register 位址	參數名稱	資料類型	存取類型	說明*1
2304 (0x0900)	緊急停止	bool	讀 / 寫	全部的軸緊急停止並解激磁。 Bit 0 1：緊急停止。
2305 (0x0901)	全部停止運動	bool	讀 / 寫	全部的軸停止運動。 Bit 0 1：全部停止運動。
2306 (0x0902)	停止運動	bool	讀 / 寫	選擇的軸停止運動。 Bit 0 1：停止運動。
2307 (0x0903)	正向吋動 (Jog+)	bool	讀 / 寫	選擇的軸往正方向吋動。 Bit 0 1：Jog+。
2308 (0x0904)	負向吋動 (Jog-)	bool	讀 / 寫	選擇的軸往負方向吋動。 Bit 0 1：Jog-。
2309 (0x0905)	相對移動	bool	讀 / 寫	選擇的軸進行相對移動。 Bit 0 1：相對移動。
2310 (0x0906)	P2P P1	bool	讀 / 寫	選擇的軸移動到 P2P 位置點 1。 Bit 0 1：移動到位置點 1。
2311 (0x0907)	P2P P2	bool	讀 / 寫	選擇的軸移動到 P2P 位置點 2。 Bit 0 1：移動到位置點 2。
2312 (0x0908)	歸原點	bool	讀 / 寫	選擇的軸執行歸原點。 Bit 0 1：歸原點。
2313 (0x0909)	儲存 User Table	bool	讀 / 寫	儲存 User Table。 Bit 0 1：儲存 User Table。

註：

*1：欲對選擇的軸進行運動控制，請先選擇軸並設定相關運動參數。

4.1.3 HMPL Task

使用者可存取 HMPL Task 參數的暫存空間，執行或停止 HMPL Task。最多可支援 64 個 HMPL Task (0~63) 的使用操作。HMPL Task 的資料暫存空間定義如下：

表.4.1.3.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明				
2336 (0x0920)	Task 開始或停止	bool	讀 / 寫	執行或停止 Task。 <table border="1" style="margin-left: 20px;"> <tr> <td>Bit 0</td> <td>0 : 停止。</td> </tr> <tr> <td></td> <td>1 : 執行。</td> </tr> </table>	Bit 0	0 : 停止。		1 : 執行。
Bit 0	0 : 停止。							
	1 : 執行。							

註：

*1 : Task N 參數的位址 = Register 位址 + N (N_{max} = 63)

4.2 Discrete Inputs

Discrete Inputs 暫存空間無定義參數，使用者可自由使用。

4.3 Input Registers

Input Registers 暫存空間提供 HIMC 監控各軸狀態、控制器資訊與 HMPL Task 狀態。預設參數分類定義如表 4.3.1，最多可支援 128 軸的運動命令。

表.4.3.1

參數分類	描述
Axis	監控各軸狀態，如運動狀態、位置回授、錯誤碼等。
Controller Information	監控控制器資訊，如控制器狀態、系統日期與時間等。
HMPL Task	監控 HMPL Task 狀態，如執行中、除錯模式、暫停執行等。

4.3.1 Axis

使用者可存取 Axis 參數的暫存空間，監控軸的運動狀態、位置回授、錯誤碼等訊息。

Axis 的資料暫存空間定義如下：

表.4.3.1.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明	單位 ^{*2}										
0 (0x0000)	運動狀態	uint32_t	唯讀	顯示第 N 軸的運動狀態。 <table border="1"> <tr> <td>Bit 0</td> <td>激磁。</td> </tr> <tr> <td>Bit 1</td> <td>移動中。</td> </tr> <tr> <td>Bit 2</td> <td>到位。</td> </tr> <tr> <td>Bit 3</td> <td>同步。</td> </tr> <tr> <td>Bit 4</td> <td>軸在群組內。</td> </tr> </table>	Bit 0	激磁。	Bit 1	移動中。	Bit 2	到位。	Bit 3	同步。	Bit 4	軸在群組內。	-
Bit 0					激磁。										
Bit 1					移動中。										
Bit 2					到位。										
Bit 3					同步。										
Bit 4	軸在群組內。														
1 (0x0001)															
2 (0x0002)															
3 (0x0003)															
4 (0x0004)															
5 (0x0005)	位置回授	float	唯讀	顯示第 N 軸的位置回授。	mm 或 deg										
6 (0x0006)															
7 (0x0007)	速度回授	float	唯讀	顯示第 N 軸的速度回授。	mm/s 或 deg/s										
8 (0x0008)															
9 (0x0009)	加速度回授	float	唯讀	顯示第 N 軸的加速度回授。	mm/s ² 或 deg/s ²										
10 (0x000A)															

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明	單位 ^{*2}
10 (0x000A)	急跳度 (jerk)	float	唯讀	顯示第 N 軸的急跳度。	mm/s ³ 或 deg/s ³
11 (0x000B)					
12 (0x000C)	CoE 錯誤碼	int32_t	唯讀	顯示第 N 軸 CoE 驅動器的錯誤訊息。	-
13 (0x000D)					
14 (0x000E)	軸錯誤碼	int32_t	唯讀	顯示第 N 軸最後的錯誤訊息。	-
15 (0x000F)					

註：

*1：第 N 軸各個參數的位址 = Register 位址 + 30 * N (N_{max} = 127)

*2：單位依 iA Studio 的組態設定，決定軸為線性單位 (mm) 或旋轉單位 (deg)。

4.3.2 Controller Information

使用者可存取 Controller Information 參數的暫存空間，監控控制器的資訊，如控制器狀態、系統日期與時間等。Controller Information 的資料暫存空間定義如下：

表.4.3.2.1

Register 位址	參數名稱	資料類型	存取類型	說明
4096 (0x1000)	控制器狀態 ^{*1}	uint32_t	唯讀	顯示控制器狀態。數值對應的狀態如下：
4097 (0x1001)				0 初始化。
				1 忙碌。
				2 同步狀態，可進行運動控制。
				3 非同步狀態，不可進行運動控制。
	4 發生錯誤。			
4098 (0x1002)	錯誤代碼 ^{*2}	uint32_t	唯讀	顯示最近一個 HIMC 錯誤代碼。
4099 (0x1003)				
4100 (0x1004)	系統日期	uint16_t	唯讀	顯示系統日期 (年)。
4101 (0x1005)				顯示系統日期 (月)。
4102 (0x1006)				顯示系統日期 (日)。
4103 (0x1007)	系統時間	uint16_t	唯讀	顯示系統時間 (時)。
4104 (0x1008)				顯示系統時間 (分)。
4105 (0x1009)				顯示系統時間 (秒)。

註：

*1：如需控制器狀態的相關資訊，請參閱《iA Studio 軟體使用手冊》1.5 節主畫面。如需控制器狀態對應的 HIMC 燈號，請參閱《HIMC 安裝指南》2.4 節 LED 指示燈。

*2：資料暫存空間內的錯誤代碼為十進制，請轉換為十六進制，在《iA Studio 軟體使用手冊》第 5 章附錄查詢相關說明。

4.3.3 HMPL Task

使用者可存取 HMPL Task 參數的暫存空間，監控 HMPL Task 的狀態。最多可支援 64 個 HMPL Task (0~63) 的使用操作。HMPL Task 的資料暫存空間定義如下：

表.4.3.3.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明											
4112 (0x1010)	Task 狀態 (Task 0)	int32_t	唯讀	顯示 Task 的狀態。											
4113 (0x1011)				<table border="1"> <tr> <td>Bit 0</td> <td>載入完成。</td> </tr> <tr> <td>Bit 1</td> <td>執行中。</td> </tr> <tr> <td>Bit 2</td> <td>開始除錯模式。</td> </tr> <tr> <td>Bit 3</td> <td>暫停。</td> </tr> <tr> <td>Bit 4</td> <td>執行錯誤。</td> </tr> <tr> <td>Bit 5</td> <td>Task 已被修改。</td> </tr> <tr> <td>Bit 6</td> <td>載入錯誤。</td> </tr> </table>	Bit 0	載入完成。	Bit 1	執行中。	Bit 2	開始除錯模式。	Bit 3	暫停。	Bit 4	執行錯誤。	Bit 5
Bit 0	載入完成。														
Bit 1	執行中。														
Bit 2	開始除錯模式。														
Bit 3	暫停。														
Bit 4	執行錯誤。														
Bit 5	Task 已被修改。														
Bit 6	載入錯誤。														

註：

*1 : Task N 參數的位址 = Register 位址 + N * 2 (N_{max} = 63)

4.4 Holding Registers

Holding Registers 暫存空間提供 HIMC 設定各軸的運動參數、存取控制器與從站 IO、存取控制器 User Table 及設定使用者自定義參數。預設參數分類定義如表 4.4.1，最多可支援 128 軸的運動命令、256 點的從站 IO、64 個 double 的 User Table，並保留 4X28672 ~ 4X40959 的記憶體區段供使用者自訂使用。

表.4.4.1

參數分類	描述
Axis	監控各軸狀態和設定各軸參數。
GPIO	控制 HIMC 的泛用輸入和泛用輸出 (GPIO)。
Slave GPIO	控制從站的泛用輸入和泛用輸出 (GPIO)。
User Table	存取 HIMC 的 User Table。
User-defined Parameters	HIMC 保留部分的資料暫存空間供使用者自訂參數使用。
無定義	部分的資料暫存空間無定義參數，使用者可自由使用。

註：

如需在資料暫存空間自訂參數，請參閱《iA Studio 軟體使用手冊》4.11 節 **Modbus Configuration Manager**。

4.4.1 Axis

使用者可存取 Axis 參數的暫存空間，監控或設定軸的運動參數、位置回授、錯誤碼等訊息。

Axis 的資料暫存空間定義如下：

表.4.4.1.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明	單位 ^{*2}
20480 (0x5000)	最大速度	float	讀 / 寫	設定或顯示第 N 軸的最大速度。	mm/s 或 deg/s
20481 (0x5001)					
20482 (0x5002)	最大加速度	float	讀 / 寫	設定或顯示第 N 軸的最大加速度。	mm/s ² 或 deg/s ²
20483 (0x5003)					
20484 (0x5004)	最大減速度	float	讀 / 寫	設定或顯示第 N 軸的最大減速度。	mm/s ² 或 deg/s ²
20485 (0x5005)					
20486 (0x5006)	平滑時間 (smooth time)	float	讀 / 寫	設定或顯示第 N 軸的平滑時間。	ms
20487 (0x5007)					
20488 (0x5008)	P2P 停留時間 (dwell time)	float	讀 / 寫	設定或顯示第 N 軸的停留時間。	ms
20489 (0x5009)					
20490 (0x500A)	P2P 位置點 1	float	讀 / 寫	設定或顯示第 N 軸的位置點 1。 註：點對點運動的位置點 1。	mm 或 deg
20491 (0x500B)					
20492 (0x500C)	P2P 位置點 2	float	讀 / 寫	設定或顯示第 N 軸的位置點 2。 註：點對點運動的位置點 2。	mm 或 deg
20493 (0x500D)					

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明	單位 ^{*2}
20494 (0x500E)	相對距離	float	讀 / 寫	設定或顯示第 N 軸的相對距離。 註：相對移動的移動距離，由系統呼叫執行相對移動。	mm 或 deg
20495 (0x500F)					
20496 (0x5010)	歸原點方法	int16_t	讀 / 寫	設定或顯示第 N 軸歸原點程序的歸原點方法。	-
20497 (0x5011)	快速歸原點速度	float	讀 / 寫	設定或顯示第 N 軸歸原點程序的快速歸原點速度。	mm/s 或 deg/s
20498 (0x5012)					
20499 (0x5013)	慢速歸原點速度	float	讀 / 寫	設定或顯示第 N 軸歸原點程序的慢速歸原點速度。	mm/s 或 deg/s
20500 (0x5014)					
20501 (0x5015)	歸原點加速度	float	讀 / 寫	設定或顯示第 N 軸歸原點程序的歸原點加速度。	mm/s ² 或 deg/s ²
20502 (0x5016)					
20503 (0x5017)	歸原點偏移量	float	讀 / 寫	設定或顯示第 N 軸歸原點程序的原點偏移量。	mm 或 deg
20504 (0x5018)					
20505 (0x5019)	歸原點逾時時間	int32_t	讀 / 寫	設定或顯示第 N 軸歸原點程序的逾時時間。	ms
20506 (0x501A)					

註：

*1：第 N 軸各個參數的位址 = Register 位址 + 30 * N (N_{max} = 127)

*2：單位依 iA Studio 的組態設定，決定軸為線性單位 (mm) 或旋轉單位 (deg)。

4.4.2 GPIO

使用者可存取 GPIO 參數的暫存空間，監控或設定 HIMC 的泛用輸入和泛用輸出，包含 8 個泛用輸入 (GPI1~GPI8) 和 8 個泛用輸出 (GPO1~GPO8)。GPIO 的資料暫存空間定義如下：

表.4.4.2.1

Register 位址	參數名稱	資料類型	存取類型	說明						
24576 (0x6000)	GPI	int32_t	唯讀	顯示 GPI (1~8) 的狀態。 <table border="1" data-bbox="887 689 1442 943"> <tr> <td>Bit 0</td> <td>0 : GPI1 無訊號。 1 : GPI1 有訊號。</td> </tr> <tr> <td>⋮</td> <td>⋮</td> </tr> <tr> <td>Bit 7</td> <td>0 : GPI8 無訊號。 1 : GPI8 有訊號。</td> </tr> </table>	Bit 0	0 : GPI1 無訊號。 1 : GPI1 有訊號。	⋮	⋮	Bit 7	0 : GPI8 無訊號。 1 : GPI8 有訊號。
Bit 0					0 : GPI1 無訊號。 1 : GPI1 有訊號。					
⋮	⋮									
Bit 7	0 : GPI8 無訊號。 1 : GPI8 有訊號。									
24577 (0x6001)										
24578 (0x6002)	GPO	int32_t	讀 / 寫	設定或顯示 GPO (1~8) 的狀態。 <table border="1" data-bbox="887 1021 1442 1274"> <tr> <td>Bit 0</td> <td>0 : GPO1 無訊號。 1 : GPO1 有訊號。</td> </tr> <tr> <td>⋮</td> <td>⋮</td> </tr> <tr> <td>Bit 7</td> <td>0 : GPO8 無訊號。 1 : GPO8 有訊號。</td> </tr> </table>	Bit 0	0 : GPO1 無訊號。 1 : GPO1 有訊號。	⋮	⋮	Bit 7	0 : GPO8 無訊號。 1 : GPO8 有訊號。
Bit 0					0 : GPO1 無訊號。 1 : GPO1 有訊號。					
⋮	⋮									
Bit 7	0 : GPO8 無訊號。 1 : GPO8 有訊號。									
24579 (0x6003)										

4.4.3 Slave GPIO

使用者可存取 Slave GPIO 參數的暫存空間，監控或設定從站的泛用輸入和泛用輸出，包含最多 256 個泛用輸入 (GPI1~GPI256) 和 256 個泛用輸出 (GPO1~GPO256)。Slave GPIO 的資料暫存空間定義如下：

表.4.4.3.1

Register 位址	參數名稱	資料類型	存取類型	說明
24592 (0x6010)	選擇 Slave	uint16_t	讀 / 寫	選擇 Slave 來對應各自的 Slave GPIO。 Slave 的設定值為 0 ~ 127。

註：

使用 Slave GPIO 前，需先設定以上參數來取得對應 Slave 的 GPIO。

表.4.4.3.2

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明						
24608 (0x6020)	Slave GPI: Channel ^{*2} 1~32	uint32_t	唯讀	顯示 GPI (1~32) 的狀態。 <table border="1" style="margin-left: 20px;"> <tr> <td>Bit 0</td> <td>0 : GPI1 無訊號。 1 : GPI1 有訊號。</td> </tr> <tr> <td>⋮</td> <td>⋮</td> </tr> <tr> <td>Bit 32</td> <td>0 : GPI32 無訊號。 1 : GPI32 有訊號。</td> </tr> </table>	Bit 0	0 : GPI1 無訊號。 1 : GPI1 有訊號。	⋮	⋮	Bit 32	0 : GPI32 無訊號。 1 : GPI32 有訊號。
Bit 0					0 : GPI1 無訊號。 1 : GPI1 有訊號。					
⋮	⋮									
Bit 32	0 : GPI32 無訊號。 1 : GPI32 有訊號。									
24609 (0x6021)										
24672 (0x6060)	Slave GPO: Channel 1~32	uint32_t	讀 / 寫	設定或顯示 GPO (1~32) 的狀態。 <table border="1" style="margin-left: 20px;"> <tr> <td>Bit 0</td> <td>0 : GPO1 無訊號。 1 : GPO1 有訊號。</td> </tr> <tr> <td>⋮</td> <td>⋮</td> </tr> <tr> <td>Bit 32</td> <td>0 : GPO32 無訊號。 1 : GPO32 有訊號。</td> </tr> </table>	Bit 0	0 : GPO1 無訊號。 1 : GPO1 有訊號。	⋮	⋮	Bit 32	0 : GPO32 無訊號。 1 : GPO32 有訊號。
Bit 0					0 : GPO1 無訊號。 1 : GPO1 有訊號。					
⋮	⋮									
Bit 32	0 : GPO32 無訊號。 1 : GPO32 有訊號。									
24673 (0x6061)										

註：

*1：第 N 個 Channel 區間參數的位址 = Register 位址 + 2 * N (N_{max} = 7)

*2: Slave GPI 及 Slave GPO 各自擁有最多 256 個 Channel，並以每 32 個 Channel 區分為 8 個區間 (Channel 1~32、33~64、...、225~256)。

4.4.4 User Table

使用者可存取 User Table 參數的暫存空間，讀取或寫入 User Table^{*1} 中的 index 數值。

User Table 提供 128 個 index^{*2}。

註：

*1：此為存取 HIMC 記憶體中的 User Table。

*2：資料類型為 float (預設) 時，可存取 128 個 index。資料類型為 double 時，可存取 64 個 index。

- 資料類型為 float 時，User Table 的暫存空間定義如下：

表.4.4.4.1

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明
24736 (0x60A0)	index 0	float	讀 / 寫	設定或顯示 User Table 中 index 0 的數值。
24737 (0x60A1)				

註：

*1：index N 參數的位址 = Register 位址 + N * 2 (N_{max} = 127)

- 資料類型為 double 時，User Table 的暫存空間定義如下：

表.4.4.4.2

Register 位址 ^{*1}	參數名稱	資料類型	存取類型	說明
24736 (0x60A0)	index 0	double	讀 / 寫	設定或顯示 User Table 中 index 0 的數值。
24737 (0x60A1)				
24738 (0x60A2)				
24739 (0x60A3)				

註：

*1：index N 參數的位址 = Register 位址 + N * 4 (N_{max} = 63)

4.4.5 User-defined Parameters

使用者可存取 User-defined Parameters 的暫存空間，讀取或寫入自定義的參數。Register 位址範圍為 4X28672 ~ 4X40959。使用者需先透過 iA Studio 定義欲存取的參數，介面如下：

Parameter	Data Type	Access Type	Value	Start Register	End Register
Coils(0X)					
Discrete Inputs(1X)					
Input Registers(3X)					
Holding Registers(4X)					
Axis					
GPIO					
HIMC GPI	int32_t	Read	0	24576	24577
HIMC GPO	int32_t	Read / Write	0	24578	24579
Slave					
User Table					
User-Defined Parameters					
Click to select or enter p...	int8_t	Read / Write	-	28672	28672

圖 4.4.5.1 Modbus Manager

註：

如需設定自定義參數的相關資訊，請參閱《iA Studio 軟體使用手冊》4.11 節 **Modbus Configuration Manager**。

4.4.6 無定義

Register 位址 4X40960 ~ 4X65535 無定義參數，使用者可自由使用。