

*ZS 伺服驅動器  
通訊說明文件*

*Rev. 0.8.9*

# RS485 通訊格式

Baud Rate: 115200 bps 傳輸率 (預設)

Parity: Even 偶 校正位

Data Bits: 8 位元 資料格式

Stop Bits: 1 位 停止位元

所有封包使用類 RTU 模式 (皆 HEX)

字符間逾時: 0.1 sec (預設)

**注意：**字符間逾時值 和 Baud Rate 可以由驅動器人機軟體 ZEROware 進行設定。

# RS485 通訊接線

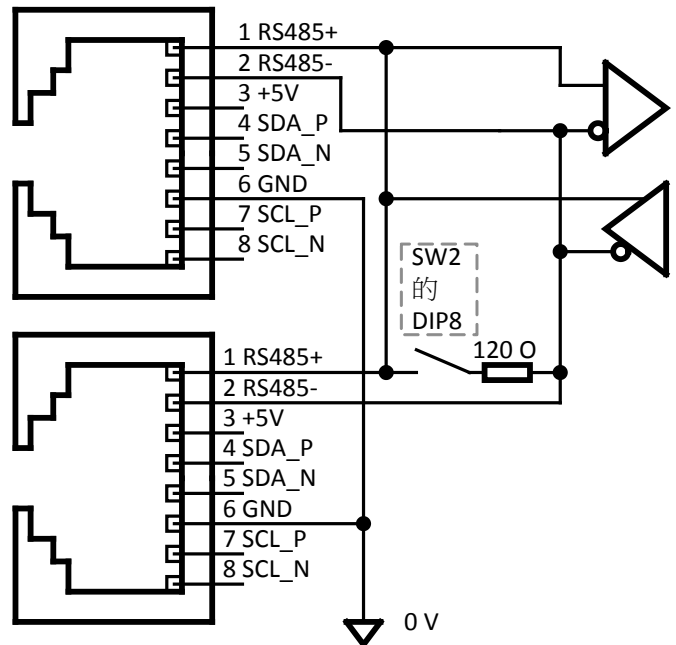
驅動器的 CN5 和 CN6 通訊接口電路完全相同。

終端電阻連接：

將指撥 SW2 的 DIP8 扳到 ON 即為連結 120Ω 之終端電阻，扳到 OFF 即為斷開 120Ω 之終端電阻。

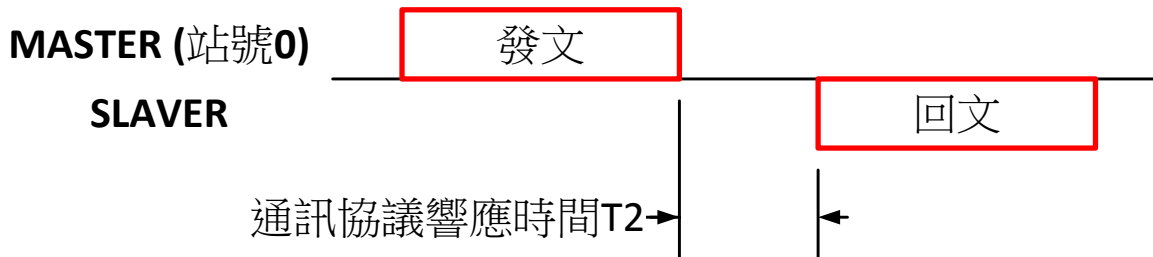
CN5 和 CN6 的腳位分配表

Pin No.	Signal Name	Description
1	RS485+	RS485 通訊正端信號(+)
2	RS485-	RS485 通訊負端信號(-)
3	+5V	內部供應的+5VDC
4	SDA_P	不使用(預留接腳)
5	SDA_N	不使用(預留接腳)
6	GND	電源地端
7	SCL_P	不使用(預留接腳)
8	SCL_N	不使用(預留接腳)

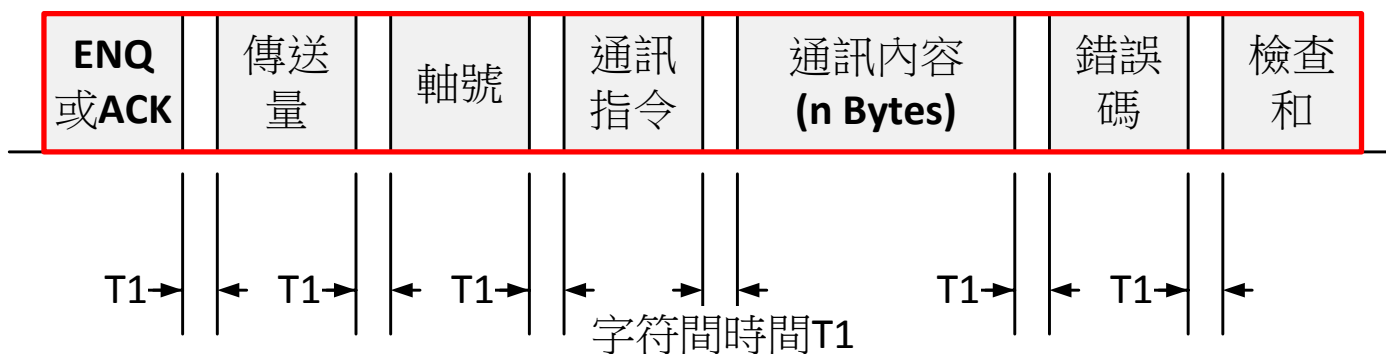


**注意：**通訊口的 6 號接腳 GND，與 CN4 的接腳 L，CN2 的電源負端並未絕緣。

# 通訊時序圖



## 發文或回文封包



- (1) 如果字符間時間  $T1$  超過字符間逾時值，則該封包會被捨棄掉，SLAVER 重新回到接收命令狀態。
- (2) 如果通訊協議響應時間  $T2$  超過通訊協議逾時值，MASTER 會重新發送封包，連續三次通訊協議響應逾時，可視為通訊異常。

# 通訊數據塊結構

通訊請求 ENQ : 0x05

通訊回應 ACK : 0x06

ENQ (0x05) 或 ACK (0x06)
傳送資料量 Byte 數
接收對象 Slaver 軸號
通訊指令
通訊內容 (n 個 Byte)
錯誤碼
檢查和 Checksum

**傳送資料量 Byte 數**：黃色框內的“通訊內容”加上“錯誤碼”的總 Byte 數 = n+1

**通訊指令**：各種要動作的代碼，詳細請見通訊指令列表。

**通訊內容**：依據不同的通訊指令，有些指令或回應會夾帶著命令或回應內容。

**錯誤碼**：回應的封包會有此代碼，檢查此碼可確認目前所下的通訊指令或通訊內容是否有錯誤發生。

**檢查和 Checksum**：計算包含“傳送資料量 Byte 數”、“接收對象 Slaver 軸號”、“通訊指令”、“通訊內容 (n 個 Byte)”、“錯誤碼”，這些所有 Byte 的總和的 2 的補碼。

範例：從 Slaver1 讀取目前警報值，讀取目前警報的通訊指令為 0x05

接收數據(Master→Slaver1)

ENQ (0x05) (通訊請求)
0x00 (傳送資料量)
0x01 (軸號)
0x05 (通訊指令)
Checksum (0xFA)

發送數據(Slaver1→Master)

ACK (0x06) (通訊回應)
0x02 (傳送資料量)
0x81 (軸號)
0x05 (通訊指令)
0xFF (目前警報代碼：無異常)
0x00 (錯誤碼)
Checksum (0xF9)

**發送端檢查和 Checksum**：(傳送資料量 0x00 + 軸號 0x01 + 通訊指令 0x05)=0x06，所以 2 的補碼值為 ( 0xFF - 0x06 + 0x01) = 0xFA。

**接收端檢查和 Checksum**：(傳送資料量 0x02 + 軸號 0x01 + 通訊指令 0x05 + 目前警報代碼 0xFF + 錯誤碼 0x00)=0x07，所以 2 的補碼值為( 0xFF - 0x07 + 0x01) = 0xF9。

**接收端回應軸號**：假設主機發命令給第 1 軸，當第 1 軸回應時會在封包內填入自己的軸號(0x01)再加上 0x80，也就是 0x81。

# 通訊指令列表

定義名	指令	指令號
<b>ccmd_STATUS</b>	取得驅動器狀態	0x00
<b>ccmd_PRM_RD</b>	參數讀取	0x01
<b>ccmd_PRM_WR</b>	參數寫入(暫時性，斷電即消失)	0x02
<b>ccmd_ALM_RD</b>	讀取目前警報	0x05
<b>ccmd_ALM_CLR</b>	清除目前所有警報	0x06
<b>ccmd_ALM_HISTORY_RD</b>	讀取警報記錄	0x07
<b>ccmd_MON_VEL_FB</b>	讀取目前回授速度 (Unit:RPM)	0x50
<b>ccmd_MON_TRQ_REF</b>	讀取目前轉矩命令 (Unit:0.1%額定轉矩)	0x51
<b>ccmd_MON_ENC_FB</b>	讀取目前編碼器回授	0x52
<b>ccmd_MON_PULSE_ACCUM</b>	讀取目前脈衝累積	0x53
<b>ccmd_MON_LATCH_POS</b>	讀取捕抓的位置點	0x54
<b>ccmd_MON_TRQ_FB</b>	讀取目前回授轉矩 (Unit:0.1%額定轉矩)	0x55
<b>ccmd_MON_VEL_FB_PRECISE</b>	讀取目前回授速度(精確) (Unit: 0.001RPM)	0x58
<b>ccmd_MON_VEL_REF_PRECISE</b>	讀取目前速度命令(精確) (Unit: 0.001RPM)	0x59
<b>ccmd_MON_DIN</b>	讀取目前輸入接點狀態	0x5A
<b>ccmd_MON_DOUT</b>	讀取目前輸出接點狀態	0x5B
<b>ccmd_MON_DIN_GPIO</b>	讀取輸入 Xm 接點的可程控 IO(GPIO)設定	0x62
<b>ccmd_MON_DOUT_GPIO</b>	讀取輸出 Ym 接點的可程控 IO(GPIO)設定	0x63
<b>ccmd_MON_LTMOD_GPIO</b>	讀取輸入 Xm 接點 GPIO 下的 LATCH 設定	0x64
<b>ccmd_GET_CONTROL_RIGHT</b>	設定控制權在通訊	0xC0
<b>ccmd_RELEASE_CONTROL_RIGHT</b>	設定控制權在外部 IO	0xC1
<b>ccmd_DIN_GPIO</b>	指定輸入 Xm 接點為可程控 IO(GPIO)	0xC2
<b>ccmd_DOUT_GPIO</b>	指定輸出 Ym 接點為可程控 IO(GPIO)	0xC3
<b>ccmd_LTMOD_GPIO</b>	指定輸入 Xm 接點為 LATCH 模式	0xC4
<b>ccmd_HOLD</b>	停止位置或速度的運動指令	0x25
<b>ccmd_LT_CLR</b>	清除輸入 Xm 接點 LATCH 的 ON 狀態為 OFF	0x29
<b>ccmd_DOUT_ON</b>	指定輸出 Ym 接點為 ON	0x2A
<b>ccmd_DOUT_OFF</b>	指定輸出 Ym 接點為 OFF	0x2B
<b>ccmd_DOUT_ON_OFF</b>	指定輸出 Ym 接點為 ON 或 OFF	0x2C
<b>ccmd_SVON</b>	伺服啟動	0x31
<b>ccmd_SVOFF</b>	伺服解除	0x32
<b>ccmd_POSING</b>	增量位置定位	0x35
<b>ccmd_POSING_ABS</b>	絕對位置定位	0x36
<b>ccmd_MHOME_SEARCH</b>	機械原點復歸	0x37
<b>ccmd_EHOME_RET</b>	電氣原點復歸	0x38
<b>ccmd_VELCTRL_EXT</b>	速度控制(使用外部類比)	0x39
<b>ccmd_VELCTRL_INT</b>	速度控制(使用通訊速度命令)	0x3A

<i>ccmd_TRQCTRL_EXT</i>	轉矩控制(使用外部類比)	0x3B
<i>ccmd_TRQCTRL_INT</i>	轉矩控制(使用通訊轉矩命令)	0x3C

通訊指令 0x00 • *ccmd\_STATUS* 取得驅動器狀態

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x00 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x00 (通訊指令)
驅動器狀態 L
驅動器狀態 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

驅動器狀態 L

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 未到達	0: 未到位	0: 未供應	0: 不可進行	0: 未伺服啟動	保留	0: 正常	0: 正常
1: 到達轉矩 限制	1: 位置已到 位	1: 動力電正常 供應	1: 可進行伺服啟 動	1: 已經伺服啟 動		1: WARN	1: ALARM

驅動器狀態 H

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	0: 未到達	0: 未到達	0: 未到達	0: 未檢出	0: 未檢出
			1: 到達欲檢 測速度	1: 到達欲檢 測轉矩	1: 瞬間到達 轉矩限制	1: CW 極限 開關檢出	1: CCW 極限 開關檢出

通訊指令 0x01

• *ccmd\_PRM\_RD* 參數讀取

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x01 (通訊指令)
參數通訊碼
Checksum

發送數據(Slaver→Master)

ACK (0x06)
A+2
Slaver 軸號+0x80
0x01 (通訊指令)
參數通訊碼
參數內容(L 到 H 共 A bytes)
L
---
H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	參數通訊碼錯誤			
1: 有錯誤發生							

通訊指令 0x02

• *ccmd\_PRM\_WR* 參數寫入(暫時性，斷電即消失)

接收數據(Master→Slaver)

ENQ (0x05)
A+1
接收對象 Slaver 軸號
0x02 (通訊指令)
參數通訊碼
參數內容(L 到 H 共 A bytes)
L
----
H
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x02
Slaver 軸號+0x80
0x02 (通訊指令)
參數通訊碼
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	參數內容 錯誤	通訊指令錯誤	RS485 錯誤	參數通訊碼 錯誤			
1: 有錯誤發生							



通訊指令 0x05

• *ccmd\_ALM\_RD* 讀取目前警報

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x05 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x02
Slaver 軸號+0x80
0x05 (通訊指令)
目前警報代碼
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	參數通訊碼錯誤			
1: 有錯誤發生							

警報代碼分為 **ALARM** 等級與 **WARN** 等級，**WARN** 等級可以即時清除，**ALARM** 等級需重新啟動驅動器電源才可復原。

警報代碼	閃燈次數	內容解說
0xFF	不閃	無任何警報
0x00	5	<b>ALARM</b> 等級，過電流保護。
0x01	5	<b>ALARM</b> 等級，功率模塊保護。
0x02	8	<b>ALARM</b> 等級，回授異常保護。
0x03	恆亮	<b>ALARM</b> 等級，系統運作錯誤保護。
0x04	5	<b>ALARM</b> 等級，過熱保護。
0x80	3	<b>WARN</b> 等級，低電壓保護。
0x81	3	<b>WARN</b> 等級，過電壓保護。
0x83	6	<b>WARN</b> 等級，速度過快保護。
0x84	4	<b>WARN</b> 等級，追隨落後保護。
0x86	2	<b>WARN</b> 等級，過負載保護。
0x87	1	<b>WARN</b> 等級，過行程保護。
0x88	7	<b>WARN</b> 等級，系統參數錯誤保護。
0x89	3	<b>WARN</b> 等級，電力缺損保護。

**通訊指令 0x06**

**• ccmd\_ALM\_CLR 清除目前所有警報**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x06 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x06 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	有無法清除之警報	通訊指令錯誤	RS485 錯誤				尚未取得控制權
1: 有錯誤發生							

**通訊指令 0x07**

**• ccmd\_ALM\_HISTORY\_RD 讀取警報記錄**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x07 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x11
Slaver 軸號+0x80
0x07 (通訊指令)
最後記錄的警報代碼
...(中間 14 個警報記錄代碼)
所記錄的最舊警報代碼
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

**通訊指令 0x50**

• **ccmd\_MON\_VEL\_FB** 讀取目前回授速度

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x50 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x50 (通訊指令)
速度回授 L
速度回授 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

讀出的速度回授單位為 RPM

**通訊指令 0x51**

• 讀取目前轉矩命令 (Unit:0.1%額定轉矩)

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x51 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x51 (通訊指令)
轉矩命令 L
轉矩命令 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

讀出的轉矩命令單位為 0.1%額定轉矩

**通訊指令 0x52**

• *ccmd\_MON\_ENC\_FB* 讀取目前編碼器回授

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x52 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x09
Slaver 軸號+0x80
0x52 (通訊指令)
編碼器回授(L 到 H 共 8 bytes)
L
----
H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

**通訊指令 0x53**

• *ccmd\_MON\_PULSE\_ACCUM* 讀取目前脈衝累積

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x53 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x53 (通訊指令)
脈衝累積 L
脈衝累積 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

通訊指令 0x54

• *ccmd\_MON\_LATCH\_POS* 讀取捕抓的位置點

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x54 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x09
Slaver 軸號+0x80
0x54 (通訊指令)
編碼器回授(L 到 H 共 8 bytes)
L
---
H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

在純位置控制模式下(即 SW2 指撥 [1]、[2]、[3] 皆扳到 OFF)，CN8 接頭的 X1 接點(原模式變換 CCM 功能的接點)，一旦被正緣觸發，將會捕抓目前位置回授暫存起來，而此通訊指令即用於讀取補抓到的位置回授值。

通訊指令 0x55

• *ccmd\_MON\_TRQ\_FB* 讀取目前回授轉矩

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x55 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x55 (通訊指令)
回授轉矩 L
回授轉矩 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

讀出的回授轉矩單位為 0.1%額定轉矩

通訊指令 0x58

• *ccmd\_MON\_VEL\_FB\_PRECISE* 讀取目前回授速度(精確)

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x58 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x05
Slaver 軸號+0x80
0x58 (通訊指令)
速度回授(L 到 H 共 4 bytes)
L
----
H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

讀出的速度回授單位為 0.001RPM

通訊指令 0x59

• *ccmd\_MON\_VEL\_REF\_PRECISE* 讀取目前速度命令(精確)

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x59 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x05
Slaver 軸號+0x80
0x59 (通訊指令)
速度命令(L 到 H 共 4 bytes)
L
----
H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

讀出的速度命令單位為 0.001RPM



**通訊指令 0x5A**

**• ccmd\_MON\_DIN 讀取目前輸入接點狀態**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x5A (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x5A (通訊指令)
輸入接點狀態 L
輸入接點狀態 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

**通訊指令 0x5B**

**• ccmd\_MON\_DOUT 讀取目前輸出接點狀態**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x5B (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x03
Slaver 軸號+0x80
0x5B (通訊指令)
輸出接點狀態 L
輸出接點狀態 H
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

通訊指令 0x62

• *ccmd\_MON\_DIN\_GPIO* 讀取輸入  $X_m$  接點的可程控 IO(GPIO)設定

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x62 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x02
Slaver 軸號+0x80
0x62 (通訊指令)
GPIO 設定
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

GPIO 設定

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	0: 未設 1: X4 設為機械原點 訊號輸入	0: 未設 1: X3 設為 GPIO	0: 未設 1: X2 設為 GPIO	0: 未設 1: X1 設為 GPIO	0: 未設 1: X0 設為 GPIO

通訊指令 0x63

• *ccmd\_MON\_DOUT\_GPIO* 讀取輸出 Ym 接點的可程控 IO(GPIO)設定

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x63 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x02
Slaver 軸號+0x80
0x63 (通訊指令)
GPIO 設定
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

GPIO 設定

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	保留	0: 未設 1: Y3 設為 GPIO	0: 未設 1: Y2 設為 GPIO	0: 未設 1: Y1 設為 GPIO	0: 未設 1: Y0 設為 GPIO

**通訊指令 0x64**

• *cmd\_MON\_LTMOD\_GPIO* 讀取輸入 Xm 接點 GPIO 下的 LATCH 設定

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x64 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x02
Slaver 軸號+0x80
0x64 (通訊指令)
LATCH 設定
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

LATCH 設定

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保 留	保 留	保 留	保 留	0: 未設 1: X3 設為 LATCH	0: 未設 1: X2 設為 LATCH	0: 未設 1: X1 設為 LATCH	0: 未設 1: X0 設為 LATCH

**通訊指令 0xC0**

• *cmd\_GET\_CONTROL\_RIGHT* 設定控制權在通訊

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0xC0 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0xC0 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

**通訊指令 0xC1**

• *ccmd\_RELEASE\_CONTROL\_RIGHT* 設定控制權在外部 IO

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0xC1 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0xC1 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

**通訊指令 0xC2**

• *ccmd\_DIN\_GPIO* 指定輸入 Xm 接點為可程控 IO (GPIO)

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0xC2 (通訊指令)
設定 GPIO(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0xC2 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定 GPIO

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	0: 不設 1: X4 設為機械原點 訊號輸入	0: 不設 1: X3 設為 GPIO	0: 不設 1: X2 設為 GPIO	0: 不設 1: X1 設為 GPIO	0: 不設 1: X0 設為 GPIO

通訊指令 0xC3

• *ccmd\_DOUT\_GPIO* 指定輸出 Ym 接點為可程控 IO (GPIO)

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0xC3 (通訊指令)
設定 GPIO(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0xC3 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定 GPIO

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	保留	0: 不設 1: Y3 設為 GPIO	0: 不設 1: Y2 設為 GPIO	0: 不設 1: Y1 設為 GPIO	0: 不設 1: Y0 設為 GPIO

通訊指令 0xC4

• *cmd\_LTMOD\_GPIO* 指定輸入  $X_m$  接點為 LATCH 模式

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0xC4 (通訊指令)
設定 LATCH MODE(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0xC4 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定 LATCH MODE

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保 留	保 留	保 留	保 留	0: 不設 1: X3 設為 LATCH	0: 不設 1: X2 設為 LATCH	0: 不設 1: X1 設為 LATCH	0: 不設 1: X0 設為 LATCH

通訊指令 0x25

• *ccmd\_HOLD* 停止位置或速度的運動指令

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x25 (通訊指令)
停止運動方式
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x25 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

停止運動方式：

0x00 立即停止

0x01 減速停止



通訊指令 0x29

• *ccmd\_LT\_CLR* 清除輸入 Xm 接點 LATCH 的 ON 狀態為 OFF

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x29 (通訊指令)
要清除的 LATCH(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x29 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

要清除的 LATCH

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保 留	保 留	保 留	保 留	0: 不清除 1: 清除 X3 為 OFF	0: 不清除 1: 清除 X2 為 OFF	0: 不清除 1: 清除 X1 為 OFF	0: 不清除 1: 清除 X0 為 OFF

**通訊指令 0x2A**

• *ccmd\_DOUT\_ON* 指定輸出 Ym 接點為 ON

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x2A (通訊指令)
設定輸出(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x2A (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定輸出

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	保留	0: 不變 1: Y3 設為 ON	0: 不變 1: Y2 設為 ON	0: 不變 1: Y1 設為 ON	0: 不變 1: Y0 設為 ON

通訊指令 0x2B

• *ccmd\_DOUT\_OFF* 指定輸出 Ym 接點為 OFF

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x2B (通訊指令)
設定輸出(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x2B (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定輸出

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	保留	0: 不變 1: Y3 設為 OFF	0: 不變 1: Y2 設為 OFF	0: 不變 1: Y1 設為 OFF	0: 不變 1: Y0 設為 OFF

**通訊指令 0x2C**

• *ccmd\_DOUT\_ON\_OFF* 指定輸出 Ym 接點為 ON 或 OFF

接收數據(Master→Slaver)

ENQ (0x05)
0x01
接收對象 Slaver 軸號
0x2C (通訊指令)
設定輸出(0~7bits)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x2C (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				
1: 有錯誤發生							

設定輸出

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
保留	保留	保留	保留	0: Y3 設為 OFF 1: Y3 設為 ON	0: Y2 設為 OFF 1: Y2 設為 ON	0: Y1 設為 OFF 1: Y1 設為 ON	0: Y0 設為 OFF 1: Y0 設為 ON

**通訊指令 0x31**

• *ccmd\_SVON* 伺服啟動

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x31 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x31 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				尚未取得控制權
1: 有錯誤發生							

通訊指令 0x32

• *ccmd\_SVOFF* 伺服解除

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x32 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x32 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤				尚未取得控制權
1: 有錯誤發生							

通訊指令 0x35

• *ccmd\_POSING* 增量位置定位

接收數據(Master→Slaver)

ENQ (0x05)
0x03
接收對象 Slaver 軸號
0x35 (通訊指令)
選擇內部位置
選擇最大速度
選擇加減速度
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x35 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	參數 內容 錯誤	通訊指令錯 誤	RS485 錯誤	伺服尚未 啟動			尚未取得控制權
1: 有錯誤發生							

選擇內部位置：

0x00~0x0B: 位置增量值 1~12

選擇最大速度：

0x00~0x03: S 曲線最大速度 1~4，0xFF: 回原點最大速度

選擇加減速度：

0x00~0x03: S 曲線加減速時間 1~4，0xFF: 回原點加減速時間

通訊指令 0x36

• *cmd\_POSING\_ABS* 絕對位置定位

接收數據(Master→Slaver)

ENQ (0x05)
0x03
接收對象 Slaver 軸號
0x36 (通訊指令)
選擇內部位置
選擇最大速度
選擇加減速度
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x36 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	參數 內容 錯誤	通訊指令錯 誤	RS485 錯誤	伺服尚未 啟動			尚未取得控制權
1: 有錯誤發生							

選擇內部位置：

0x00~0x0B: 位置絕對值 1~12

選擇最大速度：

0x00~0x03: S 曲線最大速度 1~4，0xFF: 回原點最大速度

選擇加減速度：

0x00~0x03: S 曲線加減速時間 1~4，0xFF: 回原點加減速時間

通訊指令 0x37

• *ccmd\_MHOME\_SEARCH* 機械原點復歸

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x37 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x37 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	伺服 尚未 啟動			尚未 取得 控制 權
1: 有錯誤發生							

通訊指令 0x38

• *ccmd\_EHOME\_RET* 電氣原點復歸

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x38 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x38 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	伺服 尚未 啟動			尚未 取得 控制 權
1: 有錯誤發生							

**通訊指令 0x39**

**• cmd\_VELCTRL\_EXT 速度控制(使用外部類比)**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x39 (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x39 (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	伺服尚未啟動			尚未取得控制權
1: 有錯誤發生							

**通訊指令 0x3A**

**• cmd\_VELCTRL\_INT 速度控制(使用通訊速度命令)**

接收數據(Master→Slaver)

ENQ (0x05)
0x04
接收對象 Slaver 軸號
0x3A (通訊指令)
通訊速度命令(L 到 H 共 4 bytes)
L
----
H
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x3A (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	參數內容錯誤	通訊指令錯誤	RS485 錯誤	伺服尚未啟動			尚未取得控制權
1: 有錯誤發生							

通訊速度命令：單位為 0.001 RPM



**通訊指令 0x3B**

**• cmd\_TRQCTRL\_EXT 轉矩控制(使用外部類比)**

接收數據(Master→Slaver)

ENQ (0x05)
0x00
接收對象 Slaver 軸號
0x3B (通訊指令)
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x3B (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常		通訊指令錯誤	RS485 錯誤	伺服尚未啟動			尚未取得控制權
1: 有錯誤發生							

**通訊指令 0x3C**

**• cmd\_TRQCTRL\_INT 轉矩控制(使用通訊轉矩命令)**

接收數據(Master→Slaver)

ENQ (0x05)
0x02
接收對象 Slaver 軸號
0x3C (通訊指令)
通訊轉矩命令 L
通訊轉矩命令 H
Checksum

發送數據(Slaver→Master)

ACK (0x06)
0x01
Slaver 軸號+0x80
0x3C (通訊指令)
錯誤碼
Checksum

錯誤碼：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: 正常	參數內容錯誤	通訊指令錯誤	RS485 錯誤	伺服尚未啟動			尚未取得控制權
1: 有錯誤發生							

通訊轉矩命令：單位為 0.1% 額定轉矩