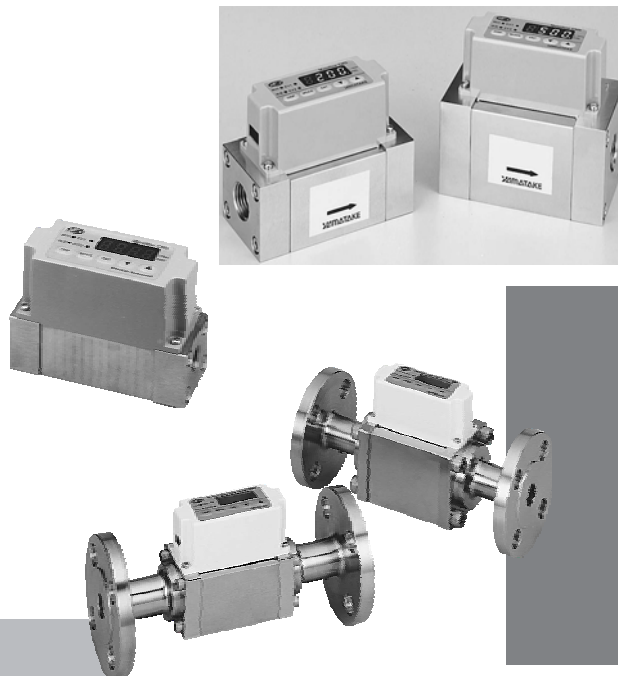




氣體質量流量計 CMS/CMF 系列 使用說明書 通訊功能篇



非常感謝您購買氣體質量流量計 CMS/CMF 系列。本使用說明書中記述了正確安全使用 CMS/CMF 系列通訊功能的必要事項。

對於承擔使用 CMS/CMF 系列通訊功能的操作盤、裝置的設計、維護的工作人員請務必在閱讀理解本書的基礎上使用。

此外，本使用說明書不只在安裝時，在維護和故障維修時也是必不可少的。請常備此手冊以供參考。

使用上的限制事項

本產品是在一般設備上使用前提下開發、設計和製造的。
在有下列安全性要求的場合應用時，請在周全考慮了事故保全設計，冗餘設計及定期維護檢查等系統和設備整體的安全性的情況下使用。

- 以人體保護為目的的安全裝置
- 輸送設備的直接控制(運行停止等)
- 航空設備
- 航天宇宙設備
- 原子能設備等

請勿把該產品用在與人身安全直接相關的用途上。

重要事項

通訊中頻繁寫入數據的場合，請對RAM的地址進行寫入。

寫入EEPROM的場合，寫入次數有限制。限制次數是10 萬次。

另外，寫入RAM的場合，當CMS/CMF系列停電時，RAM的數據將消失，回到EEPROM上的數據。

要求

請確保把本使用說明書送到本產品使用者手中。

禁止擅自複印和轉載本使用說明書的全部或部分內容。今後內容變更時恕不事先通知。

本使用說明書的內容，經過仔細審查校對，萬一有錯誤或遺漏，請向本公司提出。

對客戶應用結果，本公司有不能承擔責任的場合，敬請諒解。

©2005 Yamatake Corporation ALL RIGHTS RESERVED

μ F™、Microflow™、Microprocessor™是株式會社山武的註冊商標。

安全注意事項

■ 圖示說明

本安全注意事項的目的：爲了正確安全使用本產品，防患于未然以免給您及他人造成人身損害及財產損失，請務必遵守本安全注意事項。

本書中使用了各種圖形符號。




其顯示的含義見下所示，請認真理解所述內容。



注意

當錯誤使用本產品時，可能會造成使用者輕傷或財物損失的危險情況。

■ 圖示例

	△ 記號：在有明顯地誤操作或誤使用情況下，可能發生危險時，使用△符號表示。圖中有具體注意內容(左圖表示注意觸電)。
	⊘ 記號：爲了避免危險發生，禁止某些特定行爲時使用的符號。在圖中或在其附近注明具體禁止事項(左圖表示禁止分解)。
	● 記號：爲避免危險發生而應盡某些特定行爲的義務時使用的符號。圖中有具體指示內容(左圖是表示要把插頭從插座中拔出的意思)。

注意



請務必在切斷供給電源的情況下對本產品進行接線。
否則可能會產生故障。



請勿對本產品進行分解。
否則可能會產生故障。



請按規定的標準、指定的電線及施工方法正確配線。
否則可能造成本機故障。



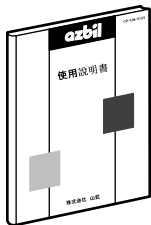
請在規格書中記載的使用條件(溫度、濕度、電壓、振動、撞擊、
安裝方向、環境等)範圍內使用本產品。
否則有發生故障的危險。



請勿讓斷線頭、鐵粉、水進入機箱內。
否則有產生誤動作或故障的危險。

本使用說明書的定位

與CMS/CMF相關的使用說明書共分5冊。根據不同的用途，請閱讀相應資料。如果您手裏無相關資料時，請向弊公司或銷售代理店索取。

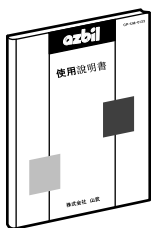


氣體質量流量計 CMS1500

資料編號 CP-UM-5171

與產品同一包裝。

用本機進行裝置設計、製作的人員，請務必閱讀。本機對使用上的安全注意事項、安裝、接線及主要規格進行說明。

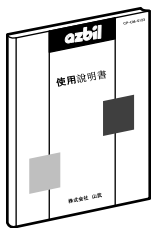


氣體質量流量計 氫氣・氮氣對應 CMS系列

資料編號 CP-UM-5207C

與產品同一包裝。

用本機進行裝置設計、製作的人員，請務必閱讀。本機對使用上的安全注意事項、安裝、接線及主要規格進行說明。

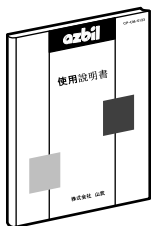


醫療用氣體儀錶 CMF0152/0155

資料編號 CP-UM-5281

與產品同一包裝。

用本機進行裝置設計、製作的人員，請務必閱讀。本機對使用上的安全注意事項、安裝、接線及主要規格進行說明。

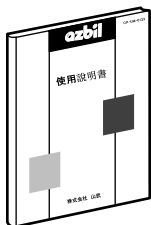


氣體質量流量計 CMS9500/0002/0005/0020/0050/0200/0500 SUS/SUS316 型

資料編號 CP-UM-5384C

與產品同一包裝。

用本機進行裝置設計、製作的人員，請務必閱讀。本機對使用上的安全注意事項、安裝、接線及主要規格進行說明。



氣體質量流量計 CMS/CMF 系列 通訊功能篇

資料編號 CP-SP-1184C

本書。

使用本機通訊功能的人員請務必閱讀。

對通訊的概要、接線、通訊步驟及本機的通訊數據一覽、故障時的處理及通訊規格進行說明。

本使用說明書的構成

本使用說明書的構成如下。

第1章 概 要

CMS/CMF 系列的通訊功能說明。

第2章 接 線

CMS/CMF 系列與其他機器間用RS-485通訊時的連接方法的說明。

第3章 設 定

CMS/CMF 系列通訊設定的說明。

第4章 通訊步驟

通訊步驟、電文的構成、數據的讀出、寫入與信號的時間的說明。

第5章 通訊數據一覽

CMS/CMF 系列通訊中使用的各種數據的地址一覽。

第6章 主站用通訊程序例

CMS/CMF 系列製作通訊程序時的注意事項及程序例的說明。

第7章 故障處理

CMS/CMF 系列的通訊不正常動作時的檢查要點的說明。

第8章 通訊規格

CMS/CMF 系列的通訊規格的說明。

附 錄

代碼表。

目 錄

安全注意事項
本使用說明書的定位
本使用說明書的構成
本使用說明書的標記

第1章 概 要	1-1
■ 特 長	1-1
■ 構成例	1-1
第2章 接 線	2-1
第3章 設 定	3-1
第4章 通訊步驟	
4-1 通訊步驟及電文的概要	4-1
■ 通訊步驟	4-1
■ 電文的構成	4-1
■ 數據鏈層	4-2
■ 應用層	4-4
4-2 命令的說明	4-5
■ 連續數據地址讀出命令 (RS 命令)	4-5
■ 連續數據地址寫入命令 (WS 命令)	4-6
4-3 應用層的數值表示	4-7
4-4 結束代碼一覽	4-8
■ 正常及警告結束	4-8
■ 異常結束	4-8
4-5 時間規格	4-9
■ 命令電文、應答電文時間規格	4-9
■ RS-485 驅動控制時間規格	4-9

第5章 通訊數據地址一覽

5-1	使用通訊數據的預備知識	5-1
■	通訊數據的種類及形式	5-1
■	通訊數據的存儲內存	5-1
■	數據地址	5-2
■	數據的單位·小數點位置	5-2
5-2	通訊數據地址一覽	5-3
■	機器種類閔連數據	5-3
■	動作狀態閔連數據	5-4
■	瞬時流量閔連數據	5-5
■	累計流量閔連數據	5-5
■	功能設定閔連數據	5-6
■	參數設定閔連數據	5-8

第6章 主站用通訊程序例

■	程序執行前	6-1
■	程序執行	6-1
■	數據讀出/寫入例子程序	6-2

第7章 故障處理

■	不能通訊時的確認項目	7-1
---	------------------	-----




第8章 通訊規格

附 錄

■	代碼表	附-1
■	與CMC10L的連接	附-2

本使用說明書的標記

本使用說明書的標記如下。

-  使用上的注意事項：表示在使用時敬請注意的事項。
-  參考：知道該項內容後助於理解。
- ：表示敬請參照的項目和頁碼。
- ①②③：表示操作步驟或者對圖進行相應說明的部分
- 》：表示操作結果及操作後的狀態。
- 、：表示顯示部的7段顯示。
- [MODE]鍵：表示顯示部的按鍵。

第 1 章 概 要

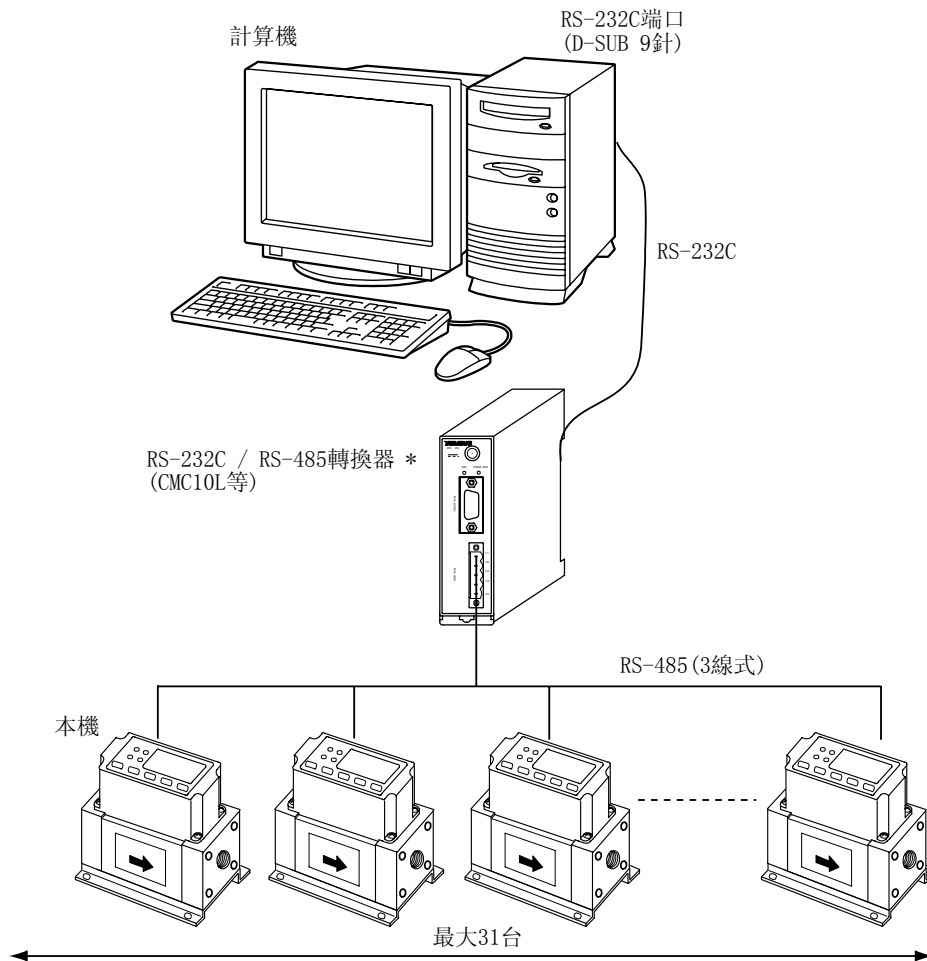
CMS/CMF系列(稱為本機或從站)可通過通訊與計算機或PLC等上位機器(稱為主站)進行設定值或數據等的交換。另外,通訊程序可由用戶製作。

■ 特 長

本機的通訊功能有如下特長。

- 對1台作為主站的上位機器,最大可連接31台本機。
- 上位機器的通訊規格為RS-232C的場合,需要使用另售的通訊轉換器CMC10L,可實現RS-232C ⇔ RS-485的轉換。
- 傳輸速度最大可對應9600bps。

■ 構成例



*: RS-232C/RS-485轉換器,可使用本公司產CMC10L001A000。

第 2 章 接 線

⚠ 注意



請務必切斷電源後再進行接線作業。
否則可能產生故障。



請不要分解本機。
否則有產生故障的可能。



請按規定的標準、指定的電線及施工方法進行正確配線。
否則可能會產生故障。

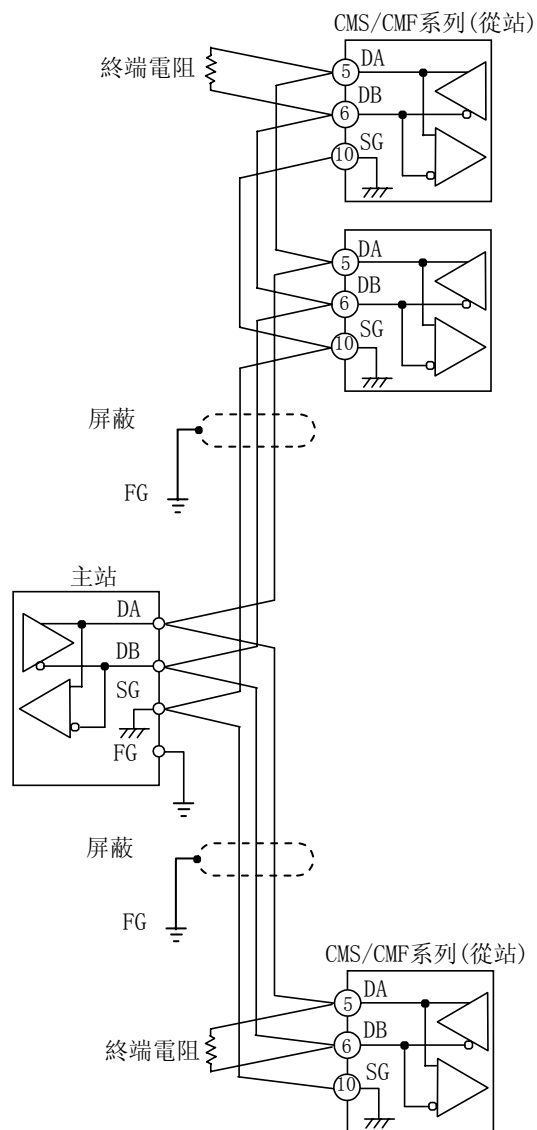


請勿讓斷線頭、鐵粉、水進入機箱內。
否則有產生誤動作、故障的危險。

📖 參考

- 關於RS-485通訊線以外的接線方法，請參閱
 - 👉 CMS1500 使用說明書 CP-UM-5171、
 - 👉 氫氣· 氮氣對應 CMS使用說明書 CP-UM-5207C、
 - 👉 CMF0152/0155 使用說明書 CP-UM-5281、
 - 👉 CMS9500/0002/0005/0020/0050/0200/0500
SUS/SUS316 型使用說明書 CP-UM-5384C。

RS-485 連接方法的一例。



通訊線路的兩端上, 請安裝 $150\ \Omega \pm 5\%$ 的 $1/2\text{W}$ 以上的終端電阻。
屏蔽的FG接地不是在屏蔽的兩端, 而是單側1點接地。
作為主站的轉換器, 可使用本公司的產品CMC10L001A000。

! 使用上的注意事項

- 請務必連接SG。如果不連接, 將不能獲得穩定的通訊。

第 3 章 設 定

爲了實現通訊功能, 請事先按下記順序對本機的通訊條件及機器地址進行設定, 使其與主站的設定相吻合。

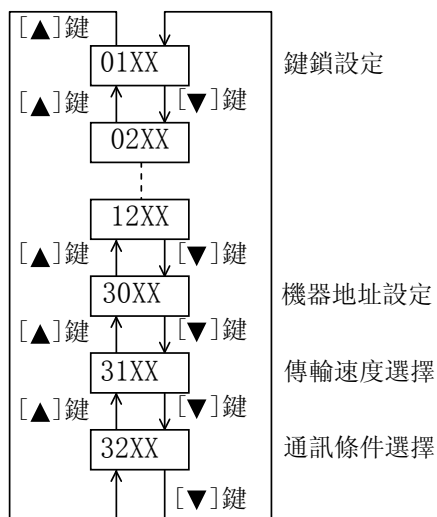
● 設定方法

按下記步驟進行功能的設定。

①按[MODE]鍵。

》顯示部的上位2位變成閃爍狀態。顯示部的上位2位是設定項目、下位2位顯示其內容。

②按[▲]鍵或[▼]鍵, 選擇希望的設定模式編號(下頁)。



③按[ENT]鍵。

》設定項目被選擇, 下位2位變成閃爍狀態。

④按[▲]鍵、[▼]鍵, 下位2位處選擇希望的設定值, 按[ENT]鍵。

》4位全部燈亮, 確定設定值。

⑤繼續設定時, 請回到①重複操作。

如果沒有其他設定的項目, 進入⑥。

⑥按[DISP]鍵。

》從功能設定模式回到瞬時流量顯示或累計流量顯示。

❗ 使用上的注意事項

- 在④的操作中(設定閃爍中)按[DISP]鍵時, 設定值不更新, 保持前次值。

● 設定項目

模式	項目內容	設定編號 及內容	初始值	參 考
30	機器地址設定	0: 不使用通訊功能 1~99: 機器地址	0	0時通訊功能不動作 請設定與其他從站不同的地址
31	傳送速度選擇	0: 9600bps 1: 4800bps 2: 2400bps	0	
32	通訊條件選擇	0: 8位數據、偶數校驗、 停止位1 1: 8 位數據、無校驗、 停止位 2	0	

第 4 章 通訊步驟

4 - 1 通訊步驟及電文的概要

在此對通訊步驟的概要及電文構成的概念進行說明。

■ 通訊步驟

通訊的步驟如下。

- ① 主站向要通訊的1台從站發送命令電文。
- ② 收到命令電文的從站, 根據電文的內容進行讀出或寫入處理。
- ③ 然後, 從站把根據處理內容的電文作為應答電文返信。
- ④ 主站接收應答電文。

■ 電文的構成

電文構成如下。

電文大致分為數據鏈層、應用層。

- 數據鏈層
含有通訊必要的基本信息的層。含有命令電文的去向或電文的檢查信息。
- 應用層
讀出寫入數據的層。內容根據目的的不同而不同。

電文由下圖的①~⑨構成。

應用層中, 存儲來自主站的命令電文的命令和來自子局的應答電文的響應。



- | | |
|-------------------|-----------------|
| ① STX(電文的起始) | ⑥ ETX(命令/響應的結束) |
| ② 機器地址 | ⑦ 檢查和 |
| ③ 子地址(00固定) | ⑧ CR(分隔符) |
| ④ 設備區分代碼 | ⑨ LF(分隔符) |
| ⑤ 命令電文=命令 應答電文=響應 | |

■ 數據鏈層

● 數據鏈層的概要

數據鏈層是固定長, 各數據的位置、文字數都有規定。

但ETX以後的數據鏈層的數據位置按應用層的文字數移動, 但文字長不變化。

● 應答開始條件

只有當數據鏈層的電文構成(機器地址、子地址、檢查和、1楨的電文長等)全部正確の場合, 從站才發送應答電文。其中任一個發生錯誤の場合, 應答電文不被發送, 處于STX收信等待狀態。

● 數據鏈層的數據定義一覽

以下是數據鏈層的數據定義一覽。

數據名	字符串代碼	字節數	數據的意味
STX	02H	1	電文的起始
機器地址	用16進制數的字符串代碼表示	2	通訊對象機器的區別
子地址	用16進制數的字符串代碼表示	2	固定為00
設備區分代碼	"X"(58H)或"x"(78H)	1	機器的種別
ETX	03H	1	應用層的結束
檢查和	把00H~FFH用2位16進制數的字符串代碼表示	2	電文的檢查和
CR	0DH	1	電文的結束(1)
LF	0AH	1	電文的結束(2)

● 數據的說明

● STX(02H)

從站收到STXの場合, 判斷為發送電文的起始。

所以, 不管之前處于何種接收狀態, 都回到初始狀態, 收到第1文字為STX的電文後開始處理。這是考慮到由于干擾等發生命令電文異常の場合, 從主站側的下次正確的電文(例如, RETRY電文)開始, 恢復從站的應答。

- 機器地址

從站根據主站的命令電文, 當有自己機器地址的場合, 處理命令, 作成應答電文。機器地址=0 (30H30H) 的場合, 即使機器地址一致, 也作為無應答。另外, 從站在應答電文上加上自己的機器地址回信。

機器地址用16進制數的2位字符串表示。

例: 通訊對象的機器地址為10時

① 10 (10進制數) = 0AH (16進制數)

② 轉換成字符串代碼

0 = 30H

A = 41H

③ ②得出的「0A」(30H 41H) 是機器地址。

- 子地址

由于本機不使用子地址, 設定為“00” (30H 30H)。從站在應答電文上加上子地址回信。

- 設備區分代碼

本機的設備區分代碼設定為X (58H) 或x (78H)。根據機器系列進行規定, 不能使用其他的文字。從站在應答電文上加上與接收電文相同的設備區分代碼後回信。初次使用X (58H), 為了區分再送電文, 使用x (78H) 很方便。

- ETX

ETX表示應用層的結束。

- 檢查和

校驗電文在通訊途中是否因為某種異常 (例如幹擾) 而未產生變化的值。

・ 檢查和的作成方法

① 從STX到ETX的電文按1字節單位進行加算。

② 加算結果取2的補碼。

③ 轉換成字符串代碼。

例: 以P4-1的電文例求出。

① 從STX到ETX的字符串代碼按每1字節加算
計算結果的下位1字節是76H。

② 76H的2的補碼為8AH。

③ 把8AH轉換成字符串代碼, 結果為(38H) 與(41H)。

- CR/LF

表示電文的最後。LF接收結束後立即變成接收電文處理的許可狀態。

■ 應用層

● 應用層的構成

應用層的構成如下。

項目	內容
命令	"RS" (10進制數形式的連續數據地址讀出命令)
	"WS" (10進制數形式的連續數據地址寫入命令)
數據分隔符	RS, WS: ", "(逗號)
數據地址	RS, WS: "501W"等
讀出數	"1"等的字符串表示的數值
寫入數值	RS, WS: "100"等字符串表示的數值

● 1楨可訪問的數據數

種類	命令的說明	RAM領域	EEPROM領域
RS	10 進制數形式的讀出命令	8	8
WS	10 進制數形式的寫入命令	4	4

4 - 2 命令的說明

■ 連續數據地址讀出命令 (RS 命令)

是讀出連續地址的數據的命令。

可從指定的讀出起始地址開始, 用一個命令就能讀出連續地址的內容。

● 命令電文

R	S	,	1	5	0	1	W	,	1
①	②		③					②	④
應用層									

- ① 命令
- ② 數據分隔符
- ③ 讀出起始地址
- ④ 讀出數據數

● 應答電文

● 正常時 (1數據讀出)

0	0	,			
①	②		③		

● 正常時 (多個數據讀出)

0	0	,								
①	②		③		②	④		②		⑤

● 異常時

X	X
①	

XX中含有結束代碼(異常)。

- ① 結束代碼*
- ② 數據分隔符
- ③ 數據1
- ④ 數據2~(n-1)
- ⑤ 數據n

*有關結束代碼的內容, 請參閱

👉 4-4 結束代碼一覽 (4-8頁)。

■ 連續數據地址寫入命令 (WS 命令)

對連續地址進行數據寫入的命令。

● 命令電文

W	S	,	1	5	0	1	W	,	1	,	6	5
①	②		③				②	④	②	⑤		

- ① 命令
- ② 數據分隔符
- ③ 寫入起始地址
- ④ 寫入數據 (第1字)
- ⑤ 寫入數據 (第2字)

● 應答電文

● 正常時

0	0
①	

● 異常/警告時

X	X
①	

XX中含有結束代碼 (異常/警告)。

- ① 結束代碼*

*有關結束代碼的內容, 請參閱

 4-4 結束代碼一覽 (4-8頁)。

4 - 3 應用層的數值表現

各數值需要進行去零表示。

以下是包含了不進行去零表示的規格。主站的命令電文必需全部發送去零表示後的數據。

● RS、WS 命令的場合

項目	規格	異常時處理
不要的空格	不可附加	中斷電文處理, 把異常結束代碼作為應答電文回信
不要的零	不可附加	
數值=0	不能省略 務必使用"0"	
其他不要的文字	表示負數時, 在數值的開始附加"-", 不能附加其他文字 正數值的場合, 不能附加"+"	
可使用的數值範圍	-32768 ~ +32767 不能超過該範圍	

4 - 4 結束代碼一覽

對應答電文, 必須返回結束代碼。

■ 正常及警告結束

結束代碼	種 別	內容及動作
00	正常	正常結束
21	警告	通訊向不可設定的地址寫入 該地址裏不會寫入任何數據, 處理繼續
23	警告	訪問了範圍外的地址, 終止讀出 訪問了範圍外的地址, 終止寫入 但之前的寫入處理已完成

■ 異常結束

結束代碼	種 別	內容及動作
40	異常	地址上未設定W 廢棄所有電文
41	異常	WS或RS未設定 廢棄所有電文
43	異常	ETX (03H) 未設定到正確的位置 地址後的「, 」未設定 廢棄所有電文
46	異常	地址異常 廢棄所有電文
47	異常	讀出字數異常 廢棄所有電文
48	異常	寫入數值異常 除異常的地址外, 其他寫入完成
99	異常	未定義命令或其他的電文異常 廢棄所有電文

4 - 5 時間規格

■ 命令電文、應答電文時間規格

有關主站的命令電文發送與從站的應答電文發送的時間規格，需要注意以下事項。

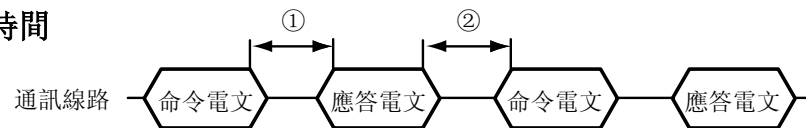
● 應答監視時間

主站發送命令電文完畢後到開始接收從站的應答電文為止的最長應答時間為2秒（①的部分）。所以，請把應答監視時間設定為2秒。通常當應答監視時間到的場合，執行命令電文的再發送。

詳見

➡ 第6章 主站用通訊程序。

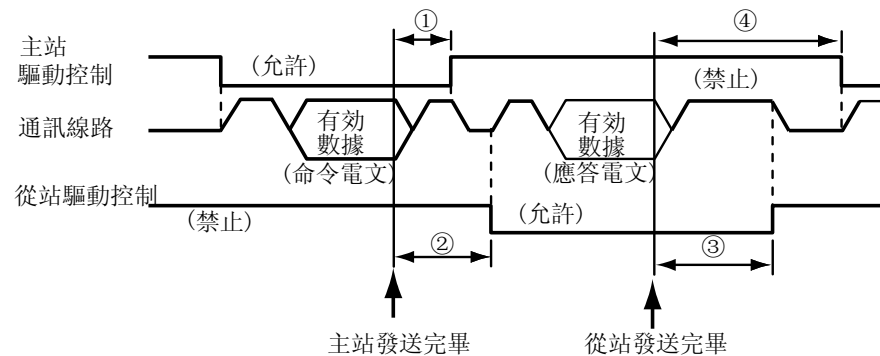
● 發送開始時間



- ① 主站發送完畢後到從站開始發送為止的時間 = 2000ms以下
(請把主站的命令電文發送完畢後的應答監視時間設定成2s)
- ② 從站完成發送後到主站發送開始為止的時間 = 50ms以上
(從主站的應答電文接收完畢後到下一個發送開始為止，請等待50ms以上)

■ RS-485 驅動控制時間規格

由主站對RS-485的發送/接收直接控制的場合，請注意以下時序。



- ① 主站發送完畢 — 驅動不允許時間 = 500 μs 以下
- ② 從站接收完畢 — 驅動允許時間 = 30ms 以上
- ③ 從站發送完畢 — 驅動不允許時間 = 10ms 以下
- ④ 主站接收完畢 — 驅動允許時間 = 50ms 以上

第 5 章 通訊數據地址一覽

5 - 1 通訊數據使用的予備知識

■ 通訊數據的種類及形式

● 通訊數據的種類

通訊數據可分為以下幾種類型。

- 機器種類閏連數據
- 動作狀態閏連數據
- 瞬時流量閏連數據
- 累計流量閏連數據
- 功能設定閏連數據
- 參數設定閏連數據

● 通訊數據的形式

通訊數據的形式劃分如下。

- 數值數據：表示數值的數據。(瞬時流量值等)
- 位數據：每個位都有含義的數據。(報警等)
位數據需要在發送時合成,接收時分解。

重要事項

對EEPROM的寫入次數小於10萬次。因為通訊需要對SP等數據進行頻繁地反復寫入的場合,請把RAM作為存儲對象。

■ 通訊數據的存儲內存

● 內存種類

通訊數據存儲在機器的內存(記憶單元)中。本機可使用的內存有2種。

- RAM：切斷電源後數據消失,但可反復寫入。
- EEPROM：切斷電源後數據也不消失。但記憶單元的特性對寫入次數有限制。
限制次數為10萬次。

● 通訊對象內存

在通訊中，對上述2種內存，有必要按目的及用途進行讀出/寫入。對象內存的差異如下。

- RAM：僅對RAM進行讀出/寫入。對RAM寫入後，電源切斷後再通電時，EEPROM內的數據複製到RAM中，RAM的數據與EEPROM相同。
- EEPROM：對RAM及EEPROM兩者進行寫入

■ 數據地址

數據地址的規定如下表所示。

通訊數據	RAM		EEPROM	
	偏移值	地址	偏移值	地址
機器種類閏連數據	1000	1001~1199	4000	4001~4199
動作狀態閏連數據	1200	1201~1399	4200	4201~4399
瞬時流量閏連數據	1400	1401~1599	4400	4401~4599
累計流量閏連數據	1600	1601~1799	4600	4601~4799
功能設定閏連數據	2000	2001~2199	5000	5001~5199
參數設定閏連數據	2200	2201~2399	5200	5201~5399

■ 數據的單位·小數點位置

讀出/寫入數據中，不附加小數點。
 單位或小數點位置由各數據決定。
 有關各數據的單位·小數點位置，請參閱機器本體的使用說明書。

5 - 2 通訊數據地址一覽

對各數據的地址及可讀出/寫入(R/W)狀態的規定如下表所示。

R/W 欄記號的含義

○：可能

×：不可

■ 機器種類関連數據

項目	數據範圍	RAM			EEPROM			備注	
		地址	R	W	地址	R	W		
各系列的氣體種類	CMS SUS/SUS316	0: 氮氣/空氣 1: 氫氣 2: 二氧化碳氣 3: 氧氣 4: 城市煤氣 13A (46MJ) 5: 沼氣 6: 丙烷 7: 丁烷 8: 用戶設定 11: 城市煤氣 13A (45MJ)	1001	○	×	4001	×	×	氣體種類設定的變更可通過功能設定(地址2008、5008)進行
	CMS 氫氣・ 氮氣對應產品	8: 用戶設定 9: 氫氣 10: 氮氣							
	CMS1500	0: 空氣/氮氣 1: 氫氣 2: 二氧化碳氣 8: 用戶設定							
	CMF	0: 氮氣/人工空氣 1: 氧氣 2: 笑氣							
未定義		1002	○	×	4002	×	×		
瞬時流量小數點 表示位置	0: 無小數點 1: xxxx. 2: xxx. x 3: xx. xx 4: x. xxx	1003	○	×	4003	×	×		
累計流量小數點 表示位置	0: 無小數點 1: xxxxxxxx. 2: xxxxxxxx. x 3: xxxxxx. xx 4: xxxxx. xxx	1004	○	×	4004	×	×		
瞬時流量單位	0: mL/min 1: L/min	1005	○	×	4005	×	×		
累計流量單位	0: mL 1: L 2: m3	1006	○	×	4006	×	×		

■ 動作狀態関連數據

項目	數據範圍	RAM			EEPROM			備注
		地址	R	W	地址	R	W	
報警狀態	(注1)	1201	○	×	4201	×	×	狀態用10進制數表示
事件狀態	(注2)	1202	○	×	4202	×	×	狀態用10進制數表示
未定義		1203	○	×	4203	×	×	
未定義		1204	○	×	4204	×	×	
累計流量下位	0~9999	1205	○	○	4205	○	○	與累計流量関連數據的地址1603及4603相同*
累計流量上位	0~9999	1206	○	○	4206	○	○	與累計流量関連數據的地址1604及4604相同*
瞬時流量	0~9999	1207	○	×	4207	×	×	與瞬時流量関連數據的地址1401相同

* : 對CMF系列，不能寫入

注 1: 報警狀態位構成(地址 1201)

0:正常 1:異常

位No.	內容
0	ALHI 超量程
1	未定義(通常為0)
2	未定義(通常為0)
3	未定義(通常為0)
4	ERR1傳感器異常
5	ERR2調整數據異常
6	ERR3傳感器加熱器異常
7	ERR4傳感器安全回路動作

注 2: 事件狀態位構成(地址 1202)

0:OFF 1:ON

位 No.	內容
0	事件輸出1狀態
1	事件輸出2狀態
2	未定義(通常為0)
3	外部接點輸入狀態
4	未定義(通常為0)
5	未定義(通常為0)
6	未定義(通常為0)
7	未定義(通常為0)

■ 瞬時流量関連數據

項目	數據範圍	RAM			EEPROM			備注
		地址	R	W	地址	R	W	
瞬時流量	0~9999	1401	○	×	4401	×	×	
事件輸出1 瞬時流量設定	0~9999	1402	○	○	4402	○	×	與參數設定的地址2201及 5201相同 *
事件輸出2 瞬時流量設定	0~9999	1403	○	○	4403	○	×	與參數設定的地址2204及 5204相同 *

* : 請用參數設定數據的地址向EEPROM執行寫入。

■ 累計流量関連數據

項目	數據範圍	RAM			EEPROM			備注
		地址	R	W	地址	R	W	
未定義		1601	○	×	4601	○	×	
未定義		1602	○	×	4602	○	×	
累計流量下位 4 位	0~9999	1603	○	○	4603	○	○	
累計流量上位 4 位	0~9999	1604	○	○	4604	○	○	
事件輸出 1 累計流量設定 下位4位	0~9999	1605	○	○	4605	○	×	與參數設定的地址2202 及5202相同 *1
事件輸出 1 累計流量設定 上位4位	0~9999	1606	○	○	4606	○	×	與參數設定的地址2203 及5203相同 *1
事件輸出 2 累計流量設定 下位4位	0~9999	1607	○	○	4607	○	×	與參數設定的地址2205 及5205相同*1
事件輸出 2 累計流量設定 上位4位	0~9999	1608	○	○	4608	○	×	與參數設定的地址2206 及5206相同 *1
反向累計流量 初期值設定 下位4位	0~9999	1609	○	○	4609	○	×	與參數設定的地址2211 及5211相同 *1 *2
反向累計流量 初期值設定 上位4位	0~9999	1610	○	○	4610	○	×	與參數設定的地址2212 及5212相同 *1 *2

*1 : 請用參數設定數據的地址向EEPROM執行寫入。

*2 : 對CMF系列，不能讀出和寫入。

■ 功能設定關連數據

項目	數據範圍	RAM			EEPROM			備注
		地址	R	W	地址	W	R	
鍵鎖設定	0: 無鍵鎖 1: 有鍵鎖	2001	○	○	5001	○	○	
測量模式設定	0: 僅瞬時流量 1: 瞬時流量/累計流量 2: 瞬時流量/反向累計流	2002	○	○	5002	○	○	對CMF系列，數據1: 固定為瞬時流量/累計流量，不可寫入
事件輸出 1 (EV1) 種類設定	0: 不使用 1: 瞬時流量上限 2: 瞬時流量下限 3: 累計流量計數增加 4: 累計流量計數減少 5: 串行輸出 6: 錯誤輸出	2003	○	○	5003	○	○	累計計數增加、反向累計數減少及累計輸出脈衝的動作的測量模式被設定為1或2時有效 累計數增加與反向累計數減少不可進行組合設定 對CMF系列，不可設定為反向累計數減少 累計流量脈衝比率根據機器種類，其值不同 ☞ 請參閱各系列的使用說明書
事件輸出 2 (EV2) 種類設定	0: 不使用 1: 瞬時流量上限 2: 瞬時流量下限 3: 累計流量計數增加 4: 累計流量計數減少 5: 累計流量脈衝比率1 6: 累計流量脈衝比率2 7: 累計流量脈衝比率3	2004	○	○	5004	○	○	
ON延遲設定 (EV1)	0: 不使用 1: 使用	2005	○	○	5005	○	○	僅當事件輸出1的設定為1或2時有效
ON延遲設定 (EV2)	0: 不使用 1: 使用	2006	○	○	5006	○	○	僅當事件輸出2的設定為1或2時有效
事件待機設定	0: 不使用 1: 使用	2007	○	○	5007	○	○	僅當事件輸出1或事件輸出2的設定為2時有效

項目	數據範圍	RAM			EEPROM			備注	
		地址	R	W	地址	R	W		
各系列的氣體種類選擇	CMS SUS/SUS316	0: 氮氣/空氣 1: 氫氣 2: 二氧化碳氣 3: 氧氣 4: 城市煤氣 13A (46MJ) 5: 沼氣 6: 丙烷 7: 丁烷 8: 用戶設定 11: 城市煤氣 13A (45MJ)	2008	○	○	5008	○	○	非氧氣對應產品, 不能指定為氧氣 ☞ 請參閱各系列的使用說明書
	CMS 氫氣・ 氮氣對應產品	8: 用戶設定 9: 氫氣 10: 氮氣							
	CMS1500	0: 空氣/氮氣 1: 氫氣 2: 二氧化碳氣 8: 用戶設定							
	CMF	0: 氮氣/人工空氣 1: 氧氣 2: 笑氣							
模擬輸出量程設定	0: 量程1 1: 量程2 2: 量程3 3: 量程4 4: 任意量程	2009	○	○	5009	○	○	根據機器種類, 量程1~4 的值不同 ☞ 請參閱各系列的使用說明書	
模擬輸出種類選擇	0: 0~5V 1: 1~5V 2: 4~20mA	2010	○	○	5010	○	○		
流量換算基準溫度	0~35	2011	○	○	5011	○	○	0~35 °C (1°C刻度)	
低流量切除設定	0: 無低流量切除 1: 小于最小顯示* 2: 小于1%FS 3: 小于2.5%FS 4: 小于5%FS	2012	○	○	5012	○	○	*根據機器種類, 最小顯示的 值不同 ☞ 請參閱各系列的使用說明書	
機器地址	0: 不使用通訊功能 1~99: 通訊地址	2030	○	×	5030	×	×		
傳送速度選擇	0: 9600bps 1: 4800bps 2: 2400bps	2031	○	×	5031	×	×		
通訊條件選擇	0: 8 位數據、 偶數校驗、停止 位1 1: 8 位數據、無校 驗、 停止位2	2032	○	×	5032	×	×		

■ 參數設定關連數據

表示	數據範圍	RAM			EEPROM			備注
		地址	R	W	地址	R	W	
事件輸出 1 瞬時流量	0~9999	2201	○	○	5201	○	○	
事件輸出 1 累計流量下位	0~9999	2202	○	○	5202	○	○	
事件輸出 1 累計流量上位	0~9999	2203	○	○	5203	○	○	
事件輸出 2 瞬時流量	0~9999	2204	○	○	5204	○	○	
事件輸出 2 累計流量下位	0~9999	2205	○	○	5205	○	○	
事件輸出 2 累計流量上位	0~9999	2206	○	○	5206	○	○	
事件輸出 1 回差	0~100	2207	○	○	5207	○	○	
事件輸出 2 回差	0~100	2208	○	○	5208	○	○	
事件輸出 1 ON延遲	0~60	2209	○	○	5209	○	○	單位(S)
事件輸出 2 ON延遲	0~60	2210	○	○	5210	○	○	單位(S)
反向累計流量 初期設定下位4位	0~9999	2211	○	○	5211	○	○	對CMF系列，不可讀出及寫入
反向累計流量 初期設定上位4位	0~9999	2212	○	○	5212	○	○	對CMF系列，不可讀出及寫入
用戶設定 氣體種類補償係數	100~8000	2213	○	○	5213	○	○	根據功能設定的氣體種類選擇，僅當選擇了用戶設定時，設定值才有效 相當于0.100~8.000
模擬輸出 量程任意設定	10~250	2214	○	○	5214	○	○	

功能設定數據與各參數設定對應的場合，設定的數據才有效

第 6 章 主站用通訊程序例

本章節所示程序例是用Borland 公司的Windows95/98/NT/2000版C++ Builder 5.0 或Borland C++ Compiler 5.5 編制的。該程序僅作為客戶編制程序時的參考，并不保證所有操作。Borland C++ Compiler 5.5可從Borland公司網站下載。

■ 程序執行前

- 請確認機器的通訊條件、機器地址。
- 從站的最長應答時間為2秒。請把主站的應答監視時間設定為2秒。2秒以內無應答時，請再次發送相同的電文。

■ 參考

- 命令電文的設備區分代碼上使用了「X」時，應答電文的設備區分代碼則為「x」。同樣，命令電文上使用「x」時，應答電文也為「X」。

主站再次發送電文時，通過交替使用設備區分代碼的「X」和「x」，可以方便的知道收到的應答電文是前次的還是本次的。

■ 執行程序

該程序執行數據的讀出及數據的寫入。執行後，顯示通訊的命令電文、應答電文的應用層。

```
命令 RS, 1000W, 2  
執行結果 00, 0, 0  
命令 WS, 1000W, 2  
執行結果 00
```

執行結果顯示例

● 執行通訊的設定

調出Open ()，初始化RS-232C。

● 執行命令

Command 中設定要執行的文字列，調出AppCPL ()。

■ 數據讀出/寫入樣板程序

❗ 使用上的注意事項

- 使用本程序例所產生的不良後果，本公司不負任何責任。

```

//-----
// C++ Builder 5
// Borland C++
// bcc32 cpl.cpp
//
// cygwin + gcc
// gcc cpl.cpp
//
#include <stdio.h>
#include <windows.h>
#pragma hdrstop

#define COMRESENDNUM 2
#define BUFFERSIZE 4096
#define TIMEOUT 2000

HANDLE handle;
unsigned long ErrorCode;
bool CheckSum;

bool AppCPL( char* tosend, char * received );
int Open( void );
int Close( void );
bool Write( unsigned char *Mesg, unsigned long Size );
bool Read( unsigned char *Buffer, unsigned long SizeToRead,
           unsigned long *ReadSize, unsigned long *ErrFlag );
void CPLSum( unsigned char *str, int len, unsigned char *buf );

//-----
int main( int argc, char* argv[] )
{
    char command[255];
    char recieve[255];

    handle = (void *)0xffffffff;
    ErrorCode = 0;
    CheckSum = true;

    if(Open()==0){
        strcpy(command,"RS,1000W,2");
        AppCPL(command,recieve);
        printf("命令 :%s\n",command);
        printf("執行結果:%s\n",recieve);

        strcpy(command,"WS,1000W,2");
        AppCPL(command,recieve);
        printf("命令 :%s\n",command);
        printf("執行結果:%s\n",recieve);
        Close();
        getchar();
    }
    return 0;
}

//-----
int Open(void)
{
    COMMTIMEOUTS Timeouts;
    _DCB DCB;

    handle = CreateFile( "¥¥¥¥.¥¥COM1", GENERIC_READ|GENERIC_WRITE,
                        0, 0, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, 0 );
    if( handle==(void *)0xffffffff ) return 3;

    if( !SetupComm( handle, BUFFERSIZE, BUFFERSIZE ) ){
        CloseHandle( handle );
        handle = (void*)0xffffffff;
        return 4;
    }
}

```

```

if( !PurgeComm( handle, PURGE_TXABORT|PURGE_RXABORT|
                PURGE_TXCLEAR|PURGE_RXCLEAR ) ){
    CloseHandle( handle );
    handle = (void*)0xffffffff;
    return 5;
}

Timeouts.ReadIntervalTimeout          = 0xFFFFFFFF;
Timeouts.ReadTotalTimeoutMultiplier  = 0;
Timeouts.ReadTotalTimeoutConstant    = 0;
Timeouts.WriteTotalTimeoutMultiplier = 0;
Timeouts.WriteTotalTimeoutConstant   = 0;

if( !SetCommTimeouts( handle, &Timeouts ) ){
    CloseHandle( handle );
    handle = (void*)0xffffffff;
    return 6;
}

if( !GetCommState( handle, &DCB ) ){
    CloseHandle( handle );
    handle = (void*)0xffffffff;
    return 7;
}

DCB.BaudRate=CBR_19200;
DCB.fBinary=1;
DCB.fParity= 1;
DCB.ByteSize=0x08;
DCB.Parity=EVENPARITY;
DCB.StopBits=ONESTOPBIT;

if( !SetCommState( handle, &DCB ) ){
    CloseHandle( handle );
    handle = (void*)0xffffffff;
    return 8;
}

return 0;
}

int Close( void )
{
    if( handle!=(void *)0xffffffff ){
        if( !SetCommMask( handle, 0 ) ){
            CloseHandle( handle );
            handle = (void*)0xffffffff;
            return 2;
        }
        if( !EscapeCommFunction( handle, CLRDTR ) ){
            CloseHandle( handle );
            handle = (void*)0xffffffff;
            return 3;
        }
        if( !CloseHandle( handle ) ) return 4;
        handle = (void*)(0xffffffff);
    }else{
        return 1;
    }

    return 0;
}

//-----
bool Write( unsigned char *Mesg, unsigned long Size )
{
    unsigned long Error;
    unsigned long Errors;
    unsigned long SizeWritten;
    COMSTAT Stat;

    if( handle==(void *)0xffffffff ){
        return false;
    }
    if( Size>BUFFERSIZE ){
        printf( " 發送數據太長      " );
        return false;
    }
}

```

```

do{
    ClearCommError( handle, &Errors, &Stat );
}while( BUFFERSIZE < Stat.cbOutQue + Size );

if( WriteFile( handle, Mesg, Size, &SizeWritten, 0 )==false ){
    if( GetLastError()==ERROR_IO_PENDING ){
        while( GetOverlappedResult( handle, 0, &SizeWritten, true ) ){
            Error = GetLastError();
            if( Error==ERROR_IO_INCOMPLETE ){
                continue;
            }else{
                ClearCommError( handle, &Errors, &Stat );
                break;
            }
        }
    }else{
        ClearCommError( handle, &Errors, &Stat );
    }
}

if( Size==SizeWritten ){
    return true;
}else{
    return false;
}
}
}
//-----
bool Read( unsigned char *Buffer, unsigned long SizeToRead,
           unsigned long *ReadSize, unsigned long *ErrFlag )
{
    BOOL          rt;
    unsigned char RecvMesg[BUFFERSIZE];
    unsigned char *cptr;
    unsigned long i;
    unsigned long stime;
    unsigned long dtime;
    unsigned long Error=0;
    unsigned long Size;
    COMSTAT       Stat;

    *ReadSize = *ErrFlag = dtime = 0;

    if( handle==(void *)0xffffffff ){
        return false;
    }

    stime = GetTickCount();
    do{
        ClearCommError( handle, &Error, &Stat );
        if( Stat.cbInQue>0 ){
            ZeroMemory( RecvMesg, BUFFERSIZE );
            rt = ReadFile( handle, RecvMesg, Stat.cbInQue, &Size, 0 );
            RecvMesg[Stat.cbInQue] = 0x00;
            if( !rt ){
                // 清除錯誤標志
                ClearCommError( handle, &Error, &Stat );
                // 設置引數錯誤標志
                *ErrFlag = Error;
                return false;
            }
        }

        for( cptr=&RecvMesg[0],i=0 ; cptr<&RecvMesg[Stat.cbInQue] ; cptr++,i++){
            if( SizeToRead==0xFFFFFFFF ){
                if( *cptr==0x02 ){
                    *ReadSize = 0;
                    Buffer[(*ReadSize)++] = *cptr;
                }else{
                    if( (*cptr=='\n') ){
                        Buffer[(*ReadSize)++] = cptr[0];
                        if( Buffer[(*ReadSize)-2]=='\r' ){
                            Buffer[(*ReadSize)] = 0x00;
                            goto OutOfWhile;
                        }
                    }
                }
                Buffer[(*ReadSize)++] = *cptr;
            }
        }
    }
}
}

```

```

        }else{
            Buffer[(*ReadSize)++] = *cptr;
            if( (*ReadSize)>=SizeToRead ){
                goto OutOfWhile;
            }
        }
    }
    Sleep(1);
    dtime = GetTickCount() - stime;
}while( (SizeToRead>*ReadSize) && (dtime<TIMEOUT) );

OutOfWhile:

if( ((SizeToRead>*ReadSize)&&(SizeToRead!=0xFFFFFFFF))||(*ReadSize==0) ){
    *ErrFlag = 0x00010000;
    return false;
}
return true;
}
//-----
void CPLSum( unsigned char *str, int len, unsigned char *buf )
{
    int          num=0;
    unsigned char *ch;

    for( ch=&str[0] ; ch<&str[len] ; ch++ ) num += *ch;
    num = ~(num&0x000000FF)&0x000000FF);
    sprintf( (char*)buf, "%02X", num );
}
//-----

bool AppCPL( char* tosend, char * received )
{
    unsigned char  theMsg[BUFFERSIZE];
    unsigned char  theApp[BUFFERSIZE];
    unsigned char  theHdr[16];
    unsigned long  SzToSnd;
    unsigned long  RdSz;
    unsigned long  ErrFlg;
    bool          rt;
    int           Cnt=0;
    unsigned char  Sum[4];
    char          *etx;

    if( handle==(void*)(0xFFFFFFFF) ){
        return false;
    }

    ZeroMemory( theMsg, BUFFERSIZE );
    sprintf( (char*)theHdr, "%x02%x30%x31%x30%x30%x58" );
    sprintf( (char*)theMsg, "%s%s%x03", theHdr, tosend );

    if( CheckSum ){
        SzToSnd = strlen( (char*)theMsg );
        CPLSum( theMsg, SzToSnd, &theMsg[SzToSnd] );
    }
    strcat( (char*)theMsg, "%r%n" );
    SzToSnd = strlen( (char*)theMsg );

resend:
    Write( theMsg, SzToSnd );
    rt = Read( theMsg, 0xFFFFFFFF, &RdSz, &ErrFlg );

    if( !rt ){
        if( Cnt++ < COMRESENDNUM ){
            goto resend;
        }
        if( ErrFlg ){
            ErrorCode = ErrFlg;
        }else{
            ErrorCode = 0x000f0000;
        }
        return false;
    }
}

```

```

}else{
    if( strcmp( (char*)theMsg, (char*)theHdr, 6 )!=0 ){
        ErrorCode = 0x00020000;
        if( Cnt++<COMRESENDNUM ){
            goto resend;
        }
        return false;
    }
    if( CheckSum ){
        CPLSum( theMsg, RdSz-4, Sum );
        if( (theMsg[RdSz-4]!=Sum[0])||(theMsg[RdSz-3]!=Sum[1]) ){
            ErrorCode = 0x00040000;
            if( Cnt++<COMRESENDNUM ){
                goto resend;
            }
            return false;
        }
    }
    ZeroMemory( theApp,BUFFERSIZE );
    CopyMemory( theApp, &theMsg[6], RdSz-6 );
    etx = strstr( (char*)theApp, "%x03" );

    if( etx==NULL ){
        ErrorCode = 0x00200000;
        if( Cnt++<COMRESENDNUM ){
            goto resend;
        }
        return false;
    }else{
        *etx = 0x00;
    }
    strcpy( received, (char*)&theApp[0] );
}
return true;
}
//-----

```


第 7 章 故障處理

■ 不能通訊時的確認項目

- ① 裝置電源是否合上。
- ② 接線有無錯誤。
- ③ 主站與本機的通訊條件設定是否一致。
下記設定項目只要有一個不同, 就不能通訊。
下劃線部分表示在本機側可設定。
傳送速度 : 9600、4800、2400bps
數據長 : 8
校驗 : 無校驗、偶數校驗
停止位 : 1 位、2 位
- ④ 主站發送的命令電文的機器地址與本機設定的地址是否一致。
出廠時, 本機的機器地址設為0, 即使命令電文的機器地址設定為00(30H、30H), 本機也不應答。
- ⑤ 多分支連接的機器, 是否設定了不同的機器地址。
- ⑥ 通訊時間與
➡4-5 時間規格 (4-9 頁)
是否相符。
- ⑦ 設備區分代碼(本機の場合, “X” 或 “x”)以外的字符串代碼, 請全部使用大寫字母。

第 8 章 通訊規格

名稱	備注
傳送形態	平衡形
傳送線路	3線式
傳送速度(bps)	9600、4800、2400
傳送距離	500m 以下 (但與MA500DIM、CMC410連接時為300m)
通訊方式	半雙工
同步方式	調步同期
數據形式	數據8位 偶數校驗 1停止位 數據8位 無奇偶校驗 2停止位
錯誤檢測	校驗檢查、校驗和
機器地址	0~99 (0の場合通訊功能無效)
連接形式	1:N (31台以下)
其他	基于RS-485

附 錄

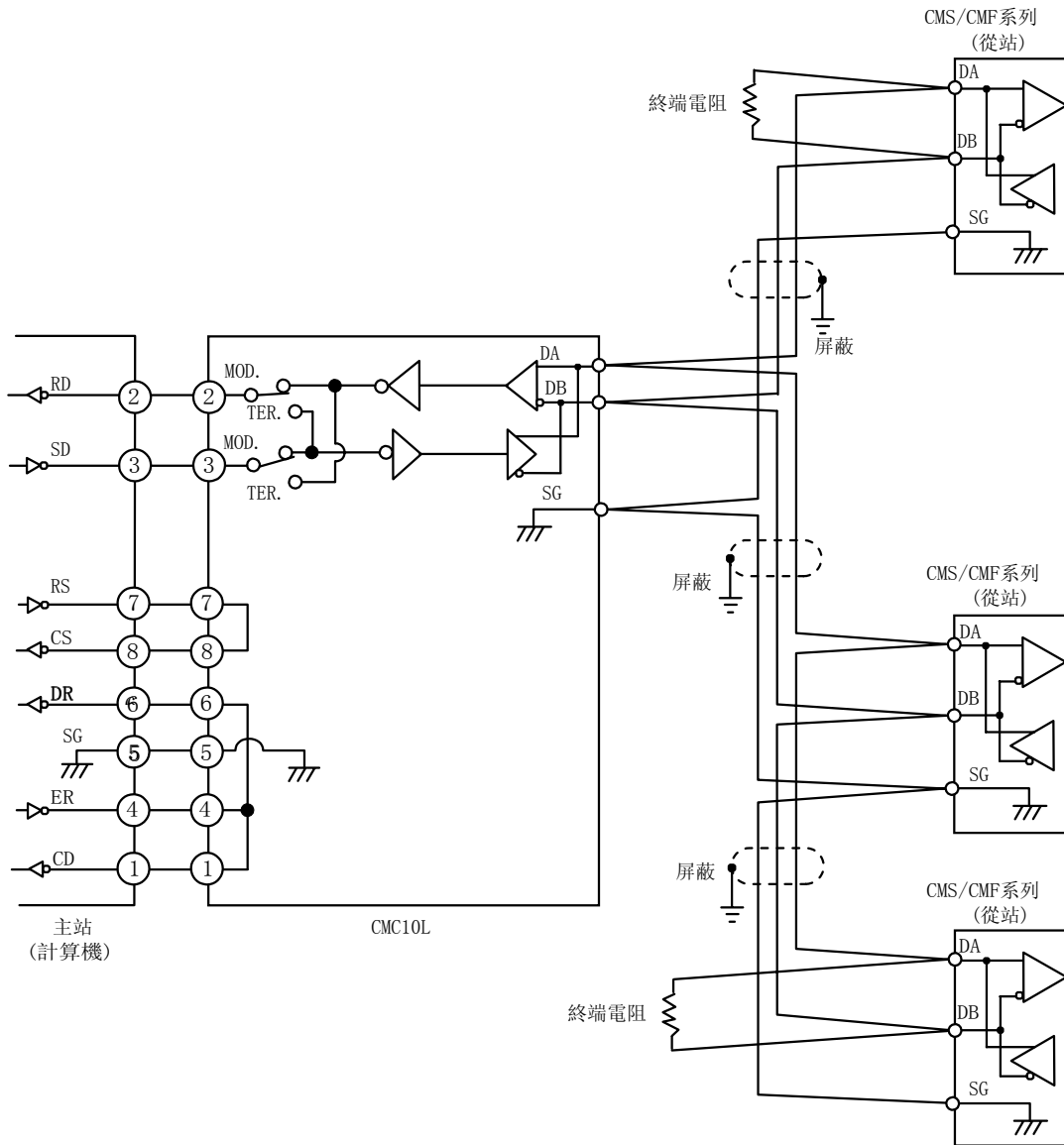
■ 代碼表

上位 下位	0	1	2	3	4	5	6	7
0			空格	0	@	P	,	p
1			!	1	A	Q	a	q
2	STX		"	2	B	R	b	r
3	ETX		#	3	C	S	c	s
4			\$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(8	H	X	h	x
9)	9	I	Y	i	y
A	LF		*	:	J	Z	j	z
B			+	;	K	[k	{
C			,	<	L	¥	l	!
D	CR		-	=	M]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

該通訊系統中不使用 部分
(根據機器, 使用的代碼不同)

■ 與CMC10L的連接

下圖是使用直通纜線D-SUB(9針)與終端模式的主站(計算機)連接的例子。



通訊線路的兩端處，請安裝 $150\ \Omega \pm 5\%$ 的 $1/2W$ 以上的終端電阻。
屏蔽的FG接地，不是在屏蔽層的兩端，而是單側的1點接地。
主站的變換器，可使用本公司生產的CMC10L001A000。

❗ 使用上的注意事項

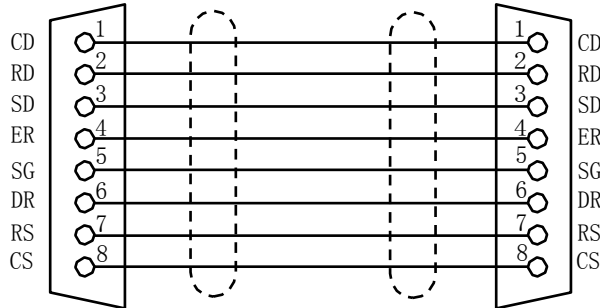
請務必連接SG。如果不連接，將不能獲得穩定的通訊。

請把主站的SD與從站的RD、主站的RD與從站的SD連接。
爲此, 主計算機(主站)側 RS-232C插口的針排列(調制解調器/終端)
要與使用的纜線種類(交叉/直通)匹配, 并按下表所示, 對CMC10L內
部的MODE開關進行設定。

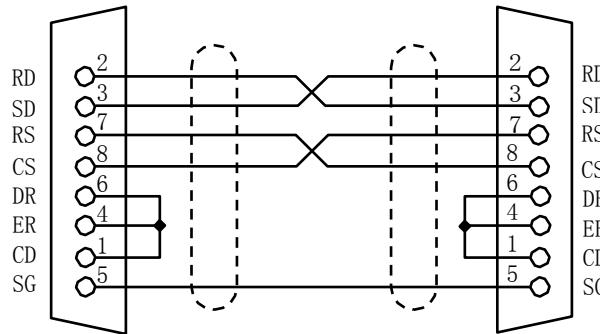
RS-232C	纜線種類	MODE開關
TERMINAL	直通	MODEM
TERMINAL	交叉	TERMINAL
MODEM	直通	TERMINAL
MODEM	交叉	MODEM

● RS-232C 纜線

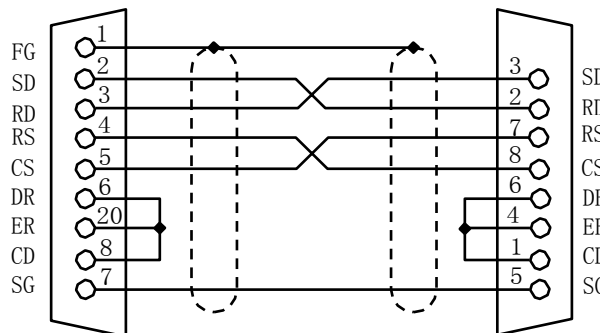
直通: 如②-②、③-③那樣, 把相同編號的針連接的兩側帶D-SUB (9針) 插口的RS-232C纜線。



交叉: 如②-③、③-②那樣, 各信號與不同的針號連接的兩側帶D-SUB (9針) 插口的RS-232C纜線。



D-SUB25針-9針轉換纜線: 如圖所示, 把D-SUB (25針) 轉換成D-SUB (9針) 的RS-232C纜線。



azbil

本資料所記內容如有變更恕不另行通知
