序列通訊概觀

何謂序列？

序列幾乎為各電腦的 標準裝置通訊協定。 請別與通訊序列匯流 排 (USB) 產 生混肴。大部分的電 腦均具有 2 組 RS232 架構的 序列埠。序列亦為通 用的通訊協定，適用 於多種裝置的儀控， 並可搭配 RS232 使用多 種 GPIB 相容 的裝置。進一步來 說，使用者可透過資 料擷取的序列溝通， 整合遠端取樣裝置。

序列溝 通的概念極為簡單。 序列埠將同時傳送並 接收 1 位元 (Bit) 的資訊 位元組 (Byte)。雖然 此傳輸量低於平行通 訊，卻可傳輸完整的 位元組；並可於較長 距離中輕鬆使用之。 舉例來說，平行通訊 的 IEEE 488 規格，必須 保持設備之間的連接 線總長度不得超過 20 公尺，任 2 個裝置之間的長 度不超過 2 公尺。而序列卻可延長 最多 1200 公 尺。

一般情況 下，工程師均使用序 列傳輸 ASCII 資料。 並透過 3 種傳輸 線完成通訊 – 接 地線 (Ground)、傳送線 (Transmit)， 與接收線 (Receive)。 由於序列為非同步 化，因此序列埠可於 其中 1 個通道傳 送資料，並於另 1 個通道接收資 料。其他通道可用於 交握，但並不為必 需。重要的序列特性 為鮑率 (Baud rate)、資料位 元、停止位元 (Stop bit)，奇偶同 位 (Parity)。 針對要溝通的 2 組通訊埠，則必須符 合這些參數：

1. 鮑率 (Baud rate) 為通訊 的速度量測作業，顯 示每秒所傳輸的位元 數。舉例來說， 300 鮑率即為每 秒 300 的位 元。工程師所稱的時 脈週期 (Clock cycle) 即為 鮑率；若協定呼叫訊 號 (Protocol call) 為 4800 鮑率，意 即時脈為 4800 Hz。
2. 亦表示序列埠以 4800 Hz 的 速率，進行資料通道 的取樣。
3. 電話線路的通用鮑率 為 14400、 28800，與 33600。鮑率當 然可以大於以上這些 數字，但這些速率將限制設備之間的距 離。因此高鮑率均用 於距離相近的裝置通 訊，常見的即為 GPIB 裝置。
4. 資料位元數，代表傳 輸作業中的實際資料 位元。當電腦傳送資 訊封包時，實際資料 總數可能不為完整 的 8 位元。資料 封包 (Data packet) 的 標準數值為 5、 7，與 8 位元。 應根據傳輸中的資 訊，選擇所需的設定。舉例來說，標 準 ASCII 具 有 0 ~ 127 的數值 (7 位元)。延伸 的 ASCII 則 使用 0 ~ 255 (8 位元)。若傳輸的資料 為簡單的純文字 (標準 ASCII)，則每 封包傳送 7 位元 資料即屬有效率的通 訊作業。1 個封包 即為單一位元組的傳 輸，包含開始/停 止 (Start/stop) 位元、資料位元，與 奇偶同位 (Parity)。由於選擇的通訊協定 將影響實際位元數 量，因此可使用「封 包 (Packet)」 代表所有的範例。
5. 停止位元 (Stop bit) 是用於針 對單一封包的通訊末 端發出訊號。常建數 值為1、1.5， 與 2 位元。由於資料將受到跨通道的 時脈所影響，而每裝 置又具有自己的時 脈，因此任 2 組 裝置可能會稍稍無法同步化。因此，停止 位元 (Stop bit) 不僅可指 示傳輸末端，並可提 供電腦時脈速度的錯誤空間。停止位元所 佔的位元數越多，則 不同時脈的同步化越 具彈性；但亦將拖慢 傳輸速度。
6. 奇偶同位(Parity) 為序列通訊錯誤檢查 的簡易形式。奇偶同 位具有4 種類 型– Even、 Odd、 Marked，與 Spaced。亦可 不使用奇偶同位。針 對Even 與 Odd 同位，序列 埠將設定同位位元 (Parity bit，為資料位元 之後的最後1 個位 元) 為1 個數 值，以確認該傳輸作 業具有邏輯高位 (Logic- high) 位元的 Even 或 Odd 數。舉例來 說，在資料為 011 的情況下， 針對Even 同位 的同位位元則為0，

以保持邏輯高位 (Logic- high) 位元的 數字平衡。在同位為 Odd 的情況下， 同位位元1 將導致 3 邏輯高位位元。 Marked 與 Spaced 同位 將不會檢查資料位 元，但會根據 Marked 同位 設定高同位位元，或 根據Spaced 同位設定低同位位 元。此將讓接收裝置 了解位元的狀態，進一步使裝置判定雜訊 是否中斷了資料，或 傳送與接收裝置是否 未同步化。

RS232 概觀

RS232 為 IBM 相容電腦架 構的序列連接方式， 可用於如連接電腦至 感測器與數據機，或用於儀器控制等許多 功能。RS232 硬體的通訊距離可達 最多 15 公尺。 RS232 並限於 進行電腦序列埠與裝置之間的點對點連 結。基於此理由，電 腦一般均需要額外 的 RS232 序 列埠。標準電腦 RS232 序列埠與許多序列介面的製 造商，將試圖平衡 Win32 API 於序列通訊 函式呼叫 (Function call) 中的功 能。Win32 API 原本是針對 數據機通訊所設計， 並未建置完整的 RS232 協定， 而無法溝通某些裝 置。

NI 則提供多種平台的 RS232 序列介 面，包含 PCI、 USB、 PCMCIA、 ExpressCard、 PXI，與乙太網路。根據所使用的平 台，NI 序列介面 具有 1、2、4、 8，與 16 埠的 版本。此外，NI RS232 序列介面更提升了某些功 能，如最高 1 Mb/s 的高速鮑 率、透過 DMA 傳輸方式的最低 CPU 使用率、選 購的 2000 V 埠對埠隔離，與 可設定的非標準鮑 率。所有的 NI 序列介面包含 NI- Serial 驅動 程式，可建置完整 的 RS232 協 定，並針對快速應用 開發提供簡單易用的 高階功能。

另請參閱：

NI 序列介面

RS422 概觀

RS422 為蘋果公司麥金塔 (Mac) 電腦的 序列連結介面。 RS422 使用差 動電子訊號，即相反 於 RS232 參照至接地的非平衡訊 號。差動傳輸使用 2 個分別用於傳輸 與接收訊號的通道， 與 RS232 相較，可達到較佳的抗 雜訊功能與較遠的距 離。較佳的抗雜訊功 能與較遠的距離，為 工業級應用的極大優勢。

NI 則提供多種平台的 RS485/RS422 序列介面，包含 PCI、USB、 PCMCIA、 ExpressCard、PXI，與乙太網 路。根據所使用的平 台，NI 序列介面 具有 1、2、4， 與 8 埠的版本。 此外，NI RS485/RS422 序列介面更提升了某 些功能，如最高 3 Mb/s 的高 速鮑率、透過 DMA 傳輸方式的 最低CPU 使用 率、選購的 2000 V 埠對 埠隔離，與可設定的 非標準鮑率。所有 的 NI 序列介面 包含 NI- Serial 驅動程式，可建置完整 的 RS485/RS422 協定，並針對快速應 用開發提供簡單易用 的高階功能。

另請參閱：

NI 序列介面

RS485 概觀

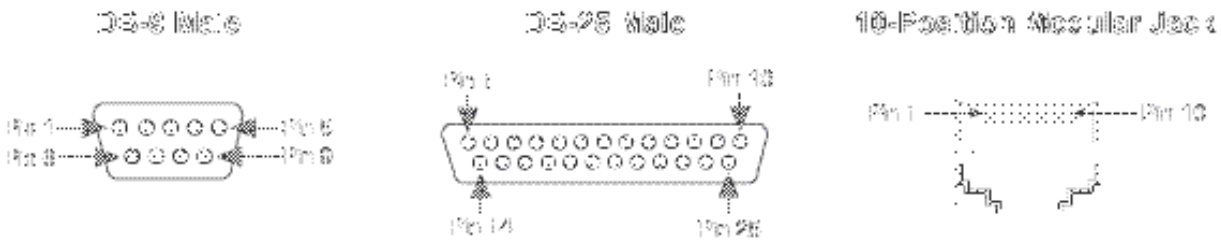
RS485 為 RS422 的改良 版本，從原本的 10 組裝置提高 至 32 組裝置， 並定義必須的電子特性，以確保最大負載 下的適當訊號電壓。 透過增強的多端點 (Multidrop) 功能，即可透過單 一 RS485 序列埠進行裝置的網路 連結。RS485 的抗雜訊與多端點功 能，使其成為工業級 應用的序列連結首選，適於將多種分散 式裝置連接至電腦或 其他控制器；用於資 料連結、HMI，與 其他作業。 RS485 為 RS422 的超集 合 (Superset)， 因此所有的 RS422 裝置可 使用 RS485 進行控制。 RS485 硬體進行的序列通訊，可達 最多約 120 公 尺的連接線長度。

NI 則提供多種平台的 RS485/RS422 序列介面，包含 PCI、USB、 PCMCIA、 ExpressCard、PXI，與乙太網 路。根據所使用的平 台，NI 序列介面 具有 1、2、4， 與 8 埠的版本。 此外，NI RS485/RS422 序列介面更提升了某 些功能，如最高 3 Mb/s 的高 速鮑率、透過 DMA 傳輸方式的 最低 CPU 使用 率、選購的 2000 V 埠對 埠隔離，與可設定的 非標準鮑率。所有 的 NI 序列介面 包含 NI- Serial 驅動程式，可建置完整 的 RS485/RS422 協定，並針對快速應 用開發提供簡單易用 的高階功能。

另請參閱：

NI 序列介面

NI 序列介面接頭的輸出 腳 (Pinouts)



DB-9 公接頭			DB-25 公接頭			10 Pin 模組 化接頭		
Pin	RS232	RS485/RS422	Pin	RS232	RS485/RS422	Pin	RS232	RS485/RS422
1	DCD	GND	2	TXD	RTS+ (HSO+)	1	無連接	無連接
2	RXD	CTS+ (HSI+)	3	RXD	CTS+ (HSI+)	2	RI	TXD-
3	TXD	RTS+ (HSO+)	4	RTS	RTS- (HSO-)	3	CTS	TXD+
4	DTR	RXD+	5	CTS	TXD+	4	RTS	RTS-

								(HSO-)
5	GND	RXD-	6	DSR	CTS- (HSI-)	5	DSR	CTS- (HSI-)
6	DSR	CTS- (HSI-)	7	GND	RXD-	6	GND	RXD-
7	RTS	RTS- (HSO-)	8	DCD	GND	7	DTR	RXD+
8	CTS	TXD+	20	DTR	RXD+	8	TXD	RTS+ (HSO+)
9	RI	TXD-	22	RI	TXD-	9	RXD	CTS+ (HSI+)
-	-	-	-	-	-	10	DCD	GND

何謂交握 (Handshaking)？

此 RS232 通訊方 式，可進行 3 種 通道的簡易連結 – Tx、Rx，與 接地。然而，針對需 要傳輸的資料，兩端均需要以相同鮑率進 行資料計時 (Clocking)。 雖然此方式適用於大 多數的應用，但卻受 限於如過載 (Overloaded) 接收器的問題回應。 此處即為序列交握 (Handshaking) 所能夠解決的地方。 3 種最為常見的 RS232 交握形 式，即為軟體交握、 硬體交握，與 Xmodem。

軟體交握

此方式將 資料位元作為控制特 性，近似於 GPIB 使用命令 字串 (Command string) 的 方式。由於控制特性可透過傳輸通道，如 正常資料般進行傳 輸，因此亦可整合 Tx、Rx，與接地 的簡單 3 線式集 合。透過 SetXMode 函式，即可啟用或停 用 2 個控制特 性：XON 與 XOFF。資料接收 器將傳送這些特性，以於通訊期間暫停傳 送器。

此方式 的最大缺點，也是最 重要的概念： Decimal 17 與 19 將 不再用於資料數值。 由於這些數值為無特 性 (Noncharacter) 數值，因此一般不會 影響 ASCII 的傳輸；然而，若以 二進位法傳輸資料，則極可能將這些數值 作為資料進行傳輸， 且傳輸作業可能發生 錯誤。

硬體交握

此方式將 使用實際的硬體通 道。如同 Tx 與 Rx 通道一 樣， RTS/CTS 與 DTR/DSR 通 道可搭配使用。若其中一個通道為輸出， 則另一個通道即為輸 入。

第一種 通道集合為 RTS (Request to Send) 與 CTS (Clear to Send)。 當接收器可接收資料 時，則將斷言 (Assert) RTS 通道以表示 準備接收資料。接著 將由 CTS 輸入 通道的傳送器讀取此 訊息，表示已可傳送 資料。

第二種 通道集合為 DTR (Data Terminal Ready) 與 DSR (Data Set Ready)。由於 此種通道可讓序列埠 與數據機溝通其狀 態，因此主要用於數 據機通訊。舉例來 說，當數據機可為電 腦傳送資料時，將先 判斷 (Assert) DTR 通道。這時 的 DTR 將顯示 由電話線所構成的連 結。接著 DSR 將讀取該訊息，電腦 則開始傳送資料。一 般常見規則，即使 用 DTR/DSR 通 道顯示系統已準備通 訊完畢，而 RTS/CTS 通 道用於個別的資料封 包。

在 LabWindows/CVI 中， SetCTSMode 函式將啟用或停用硬 體交握。若 CTS 模式為啟用 狀態， LabWindows/CVI 將依循下列規則：

電腦傳 送資料時：

RS232 程式庫 必須於傳送資料之 前，偵測該 CTS 通道為高 (High) 狀 態。

電腦接 收資料時：

若通訊 埠為開啟狀態，而輸 入佇列具有可容納資 料的空間，則程式庫 將引發 (Raise) RTS 與 DTR。

若通訊 埠輸入佇列已滿 90%，則程式庫將 降低 RTS 並提 升 DTR。

若通訊 埠輸入佇列近乎空 白，則程式庫將引 發 RTS 並保 持 DTR 為高狀 態。

若通訊 埠為關閉，程式庫將 降低 RTS 與 DTR。

XModem 交握

雖然數據 機通訊的協定極為普 遍，若其他裝置均適 用於協定，仍可直接 於裝置之間使用 XModem 協 定。在 LabWindows/CVI 中，實際的 XModem 建置 可由使用者選擇是否 隱藏。只要電腦使 用 XModem 協定連接其他裝置， 就可使用 LabWindows/CVI 的 XModem 函式傳送檔案至其他 位址 (Site)。該函 式為 XModemConfig、 XModemSend， 與 XModemReceive。

XModem 根據 下列參數使用協定： start_of_data、 end_of_trans、 neg_ack、 ack、 wait_delay、 start_delay、 max_tries， 與 packet_size。 傳輸資料的 2 邊 必須同時認可這些參 數，而 XModem 則提 供相關標準定義。然 而，使用者可透過 LabWindows/CVI 中的 XModemConfig 函式修改這些參數， 以符合任何需求。在 接收器 (Receiver) 傳送 neg_ack 特 性之後，即可於 XModem 中使 用這些參數。此特性 將告訴傳送器 (Sender) 已準備好接收資料。 接收器將於每次嘗試 之間使用 start_delay 時間，直到滿足 max_tries 或接收到傳送器的 start_of_data。 若滿足 max_tries， 則接收器將提醒使用 者目前無法溝通傳送

器。若的確接收到傳 送器的 start_of_data， 則接收器將讀取接下 來的資訊封包。此封 包具有封包號碼、作為錯誤檢查的封包補 充號碼、 packet_size 位元組的實際資料封 包，與用於更多錯誤 檢查的資料總和檢查 碼 (Checksum)。 在讀取資料之後，接 收器將呼叫 wait_delay， 並於稍後傳送認可字 元 (Ack) 回傳送器。若傳送器未 接受到 ack，則 將重新傳送 max_tries 資料封包，直到接 收 ack 為止。若傳送器一直未接 收 ack，則將通 知使用者傳送檔案失 敗。

由於傳 送器必須以 packet_size 位元組的封包傳送資 料，因此若沒有足夠 的資料填滿最後的封包，則傳送器將使 用 ASCII NULL (0) 位元組填滿資料封 包。此動作將造成已 接收的檔案大於原始檔案。由於 XModem 傳輸 的封包號碼，極可能 增加 XON/OFF 控 制的特性值而造成通 訊中斷，因此請勿 於 XModem 協定中使用 XON/XOFF。

相關連結：

NI 序列介面