

APORT 串口工作模式使用說明

一、前言

APORT 服務器擁有豐富的功能組合，每一個串口都可以指定自己的工作模式。使用者必須深入了解每一種工作模式的特性才有辦法組合出最佳方案來供自己使用，以滿足自己的應用環境。基本上我們可以支援“Real TTY COM”模式，“Real TTY GVMODEM”模式，“TCP Server”模式，“TCP Client”模式，“UDP”模式，“Multi-Serial”模式。要有如此多的模式，是因為串口的資料是以一個位元組為單位來傳送，而 IP 網路是以封包為單位來傳送，於是如何處理 IP 網路的封包傳送就有不同的特性考慮。本文嚐試說明這些特性，使用者如果能夠深入了解將對於 APORT 的使用有事半功倍之效。

二、Real TTY COM 模式

採用本模式，配合我們在 Windows NT/2000/XP/2003/Vista 及 SCO UNIX 系統的驅動程式，則我們在 APORT 的串口就如同在 PC 上面的 COM 端口一樣工作。亦即所有串口的參數（速率，資料格式，DTR 及 RTS 控制，DSR 及 DCD，CTS 的監控）都可以由應用程式來指定及監控。

由於 APORT 的串口可以分別指定給不同的 PC 服務器來使用，於是我們在安裝 Windows 驅動程式之後，可以利用 APORT_ap 這個管理程式來指定 PC 的 COM 端口號與 APORT 的串口號（我們要給定 APORT 的 IP 地址及串口號 1~16）。於是一台 PC 服務器可以指定多個 APORT 的串口為自己的 COM 端口，而同一台的 APORT 串口可以分別變成不同的 PC 服務器的 COM 端口。

基本上採用本工作模式的 COM 端口與標準 PC 的 COM 端口主要差異是在命令反應時間的延遲現象。因為我們的應用程式對標準 PC 的 COM 端口，在下命令後可以立即在硬體上反應，可是對於 APORT 的 COM 端口，應用程式的命令要由串口驅動程式變成 IP 封包再由網路驅動程式送到 APORT，再由 APORT 解析 IP 封包之後再對串口做動作。於是我們可發現其延時時間較大。

採用本模式，理論上所有針對標準 PC 的 COM 端口所寫的應用程式都可以在 APORT 的 COM 端口來執行，但如果有牽涉到實時性或對延時敏感的應用則可能會有邏輯上的問題存在。

採用本模式的好處在於 PC 服務器可以擁有串口在任何 IP 網路可以到達的地方。以前我們可能要有 MODEM 來達成此目的，現在則可以利用 IP 網路來達成此目的，而且各個 COM 端口隨時可以添加。

三、Real TTY GVMODEM 模式

採用本模式，配合我們在 UNIX/LINUX 系統的驅動程式，則我們在 APORT 的串口就如同在 UNIX 服務器上面的 TTY 端口，亦即我們的 TTY 設備端口號可以與每一個串口有固定關係存在。傳統上，UNIX 系統所支持的終端服務器其串口是採用虛終端方式來對應，亦即他是根據使用者登錄的時間先後給予設備端口號。於是每一個串口的物理位置與設備端口號無法形成一對一的唯一對應關係。在某些資料查詢的應用而言，這不會是問題，因為不同的人都可以在相同物理位置與主機系統連線進行工作而不影響系統運作。但對於某些資料庫應用而言，這是一個大問題，因為我們可能根據設備端口號來與資料庫處理鏈結，而且這些資料可能與物理位置有重大關係存在。因此我們必須令某一物理位置的串口要擁有固定且唯一的設備端口號。

採用本模式，每一個 APORT 上面的串口都可以在 UNIX 系統擁有自己唯一的設備名稱，於是應用軟件可以方便的根據設備名稱而與某一物理位置的設備連線工作。

採用本模式要注意，所有串口的參數都是要設定在 APORT 上面，因為應用軟件有關串口參數的設定並不會傳送到 APORT 上面進行修改。也就是說應用軟件與 APORT 間只有資料送收的關係，並不存在真正物理特性的關係，這是“Real TTY GVMODEM”模式與“Real TTY COM”模式最大的差異。

四、TCP Server 模式

採用本模式，APORT 上面的串口將變成 TCP 上面的端口號。於是任何人可以由任何機器利用 TCP 連接方式來與 APORT 連接。我們只要使用該串口所對應的 TCP 端口號，就可以在應用程式與串口間進行資料送收工作。

採用本模式，我們可以指定是否需要進行 Telnet 對話過程。如果我們由任何機器執行 Telnet 動作，我們就必須給定 APORT 的 IP 地址及 TCP 端口號，就可以令 Telnet 畫面與 APORT 的串口間連接起來。一般而言 Windows 的超級終端會使用 TCP 端口號 23 來進行 Telnet 動作，對於 APORT 而言，如果你用 TCP 端口號 23 來對它進行 Telnet 動作，則 APORT 會進入 Setup 畫面與動作。所以 Windows 使用者就要指定 APORT 上面串口所擁有的 TCP 端口號來進行 Telnet 才可以完成串口連線工作。

對於某些應用程式而言，它們是使用 Socket 呼叫來完成 TCP 連線方式，則我們就不執行 Telnet 對話過程，於是我們要設定在 TCP 模式而非 Telnet 模式，此時我們只是把串口資料與 TCP 封包間進行轉換。基本上本模式的串口參數是設定在 APORT 上面而不能被實時修改。任何人都可以與 APORT 連線，但是我們每一個串口只能支持一個連線，請勿兩個以上設備要同時與一個串口連接(亦即採用 TCP 要由不同 PC 與 APORT 上面同一個串口之 TCP 端口號連接)。

五、TCP Client 模式

採用本模式，APORT 上面的串口將變成 TCP 上面的端口號。於是任何串口上面的資料就會變成 TCP 封包要與指定的 PC 來連接。

由於 TCP 是一種連接模式的工作格式。當我們要送收資料之前一定要與對方連接才可以進行。可是連接需要時間，如果有串口資料才要與對方連接再進行資料送收工作，則延時太大。但如果與對方連接之後則任何資料都可以立刻與對方進行送收，可是如果對方不存在你可能不知道，所以我們在本模式存在兩個選項來指定不同工作模式。

第一個是 Startup 模式，表示 APORT 開機之後就立刻與目標 PC 機連線，只要連線建立，則資料隨時可以在串口與封包間進行轉換送收。可是在某些應用情況下，我們可能會出現與串口所連接的外部設備並未開機，但是 PC 機並不知道，於是 PC 機一直往 APORT 的串口傳資料，此時這些資料會由串口往外丟而外部設備（例如打印機）未開機，所以全部資料都被丟掉。因此我們需要某一機制來克服此問題。

第二個是 Receiving Serial data 模式。這個模式要求 APORT 開機後並不立刻與目標 PC 機連線，而是要等待串口有收到資料後才開始與目標 PC 機連線。在此模式，我們已經由串口外部設備收到資料，代表串口外部設備已經開始工作，於是我們與目標 PC 機連線之後，就可以保證由目標 PC 機往串口傳送的資料不再會因為串口外部設備未開機而變成資料丟失，做虛功的現象。

一般而言，如果我們把一台 APORT 的串口設成 TCP Server 模式，而另一台 APORT 的串口設成 TCP Client 模式，而其目標 IP 設成 TCP Server 那一台 APORT，則兩者間的串口就好像直接連接在一起一樣。一般而言，我們可以採用本結構來取代串口電纜的功用。傳統上串口電纜根據使用不同的界面標準而有不同的距離限制。例如 RS232 界面只可以連線距離在 50 feet 以內。如果要超出此距離可能要用 MODEM 等方式。現在採用本方案，則任何 IP 網絡可以到達的地方，我們都可以利用兩台 APORT 來令兩個串口設備接在一起。

六、UDP 模式

採用本模式，APORT 上面的串口將變成 UDP 上面的端口號，於是串口上面的資料就會變成 UDP 封包要與指定的 PC 來傳送。

由於 UDP 是一種無連接的格式，於是我們可以由任何地點利用 UDP 封包把資料送到 APORT 的串口指定 UDP 端口號而達成串口資料的傳送工作。而 APORT 的串口收到資料時，我們最多可以指定 4 台目標 PC 來傳送。

由於 UDP 的無連接特性，所以我們無法保證 UDP 封包可以送到目標位置，但相對的我們可以用最快速的方式送到目標位置。理論上串口電纜也不能保證你的資料一定會正確無誤送到另一端，但是如果有送到就一定是最快的速度(延時最小)，所以 UDP 模式較為接近取代串口電纜的傳輸方式。相對於 TCP 模式，由於 TCP 的連接特性，所以我們可以保證 TCP 封包（理論上如果未送到你會一直重送，除非外界強制你停止重送嘗試。）可以正確無誤送到另一端，但是你根本無法保證可以送達的時間。

因為 UDP 的特性較為接近串口電纜連線方式，所以我們也可以使用一對 APORT 都設定在 UDP 模式而且都以對方為目標 IP。於是我們就可以有最接近串口電纜的特性來完成電纜取代目的。亦即兩個串口設備間我們的電纜線不再有距離限制，只要把 APORT 成對放在兩者之間，再利用 IP 網絡來連線即可。

基本上採用本模式可以令一個串口設備同時與最多 4 台 PC 進行資料送收工作。但如果要由一台 PC 同時控制多個串口設備則要有另外的方式來達成。底下說明的“Multi-Serial”模式就是解決方案之一。

七、Multi-Serial 模式

採用本模式，APORT 上面的串口將變成 UDP 上面的端口號，於是串口上面的資料就會變成 UDP 封包要與指定的 PC 來傳送。

可是一般的 UDP 模式，每個串口都必須指定唯一的端口號，如果在同一個 APORT 盒上面對不同的串口指定相同的 UDP 端口號，則我們無法把 PC 送來的封包同時送給這些串口，因為這些串口本身執行著不同的服務程序，那一個串口的服務程序先拿走網絡來的 UDP 封包，則另一個串口就拿不到，於是無法達成某些應用。

例如我們由一台 PC 要同時向多個串口設備輪詢資料，則用傳統 UDP 模式，我們的應用程式，必須逐一向不同串口的 UDP 端口號發出封包，相當不方便，而且串口回答的封包又必須針對不同 UDP 端口號進行讀取。

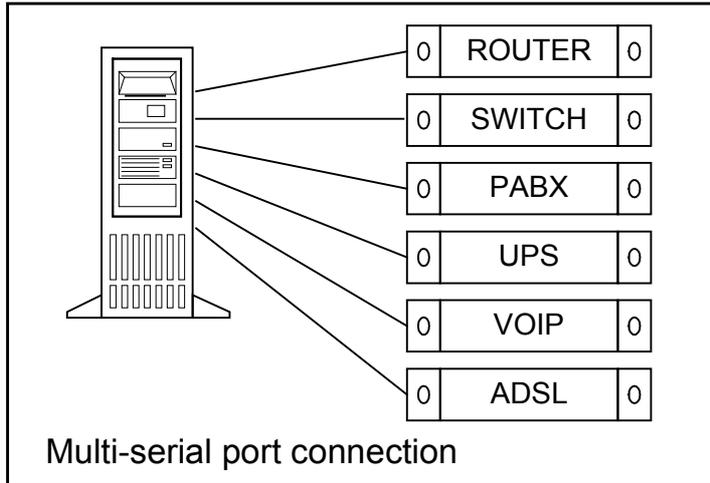
現在我們提出“Multi-Serial”模式，令同一台 APORT 上面的不同串口可以擁有相同 UDP 端口號，當我們由網絡收到此 UDP 端口號的封包資料，則我們同時向這些串口發出資料。當我們由任何串口收到資料，則可以向網絡發出 UDP 封包。所以 PC 的應用軟件撰寫相當方便，我們只要向某台 APORT 的 IP 及 UDP 端口號發出封包，就可以同時到達全部串口設備，而串口設備的回傳資料，也只用一個 UDP 端口號回傳及接收即可。

對於原本採用 RS485 網絡軟件的人，可以相當方便的採用“Multi-Serial”模式來達成利用 IP 網絡對 RS232 設備進行控制。

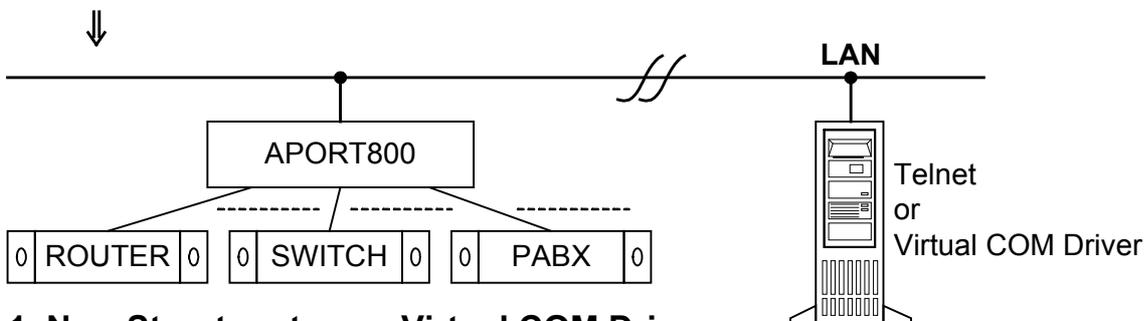
八、結論

APORT 服務器擁有不同的工作模式及不同的工作特性，所以使用者可以根據自己的需要來組合。由於我們是以每一個串口獨立設定其工作模式，因此使用者可以根據不同的應用環境來指定不同串口的工作模式。例如我們的 APORT800 擁有 8 個串口，我們可能設定串口 1~4 當作 Windows PC 的 COM 端口，我們可能設定串口 5~8 當作 UNIX PC 的 TTY 端口。同時串口 1~2 可能屬於 PC 服務器 A 的 COM 端口。而串口 3~4 可能屬於 PC 服務器 B 的 COM 端口。

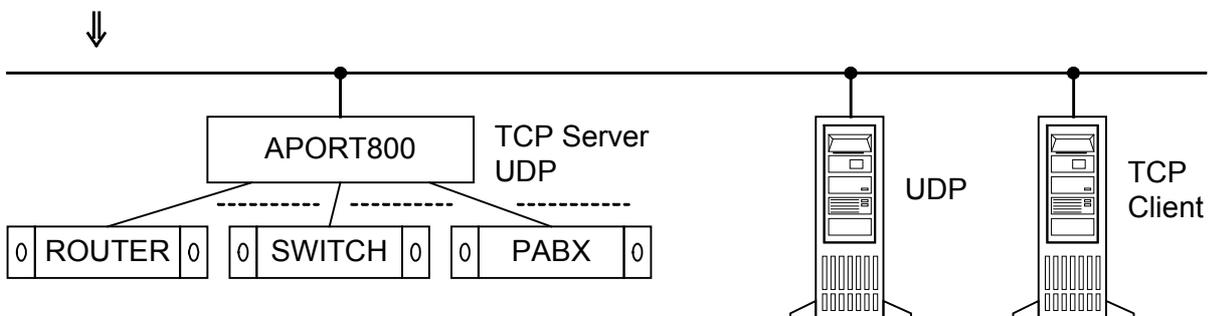
How to extend your communication distance



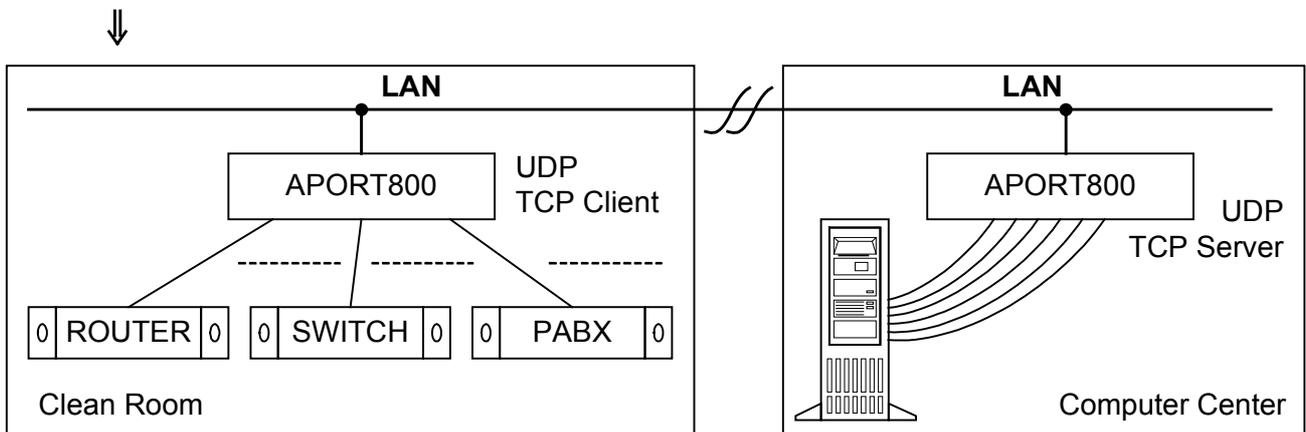
Original Structure



1. New Structure to use Virtual COM Driver.



2. New Structure to use Telnet or socket programming.



3. New Structure to use APORT800 in pair.