

# 第十七章 Token-Bus 與 Token-Ring 網路

雖然符記網路不似 Ethernet 網路的風行，但它的運作原理也值得研究網路者的探討。

## 17-1 Token Passing 網路簡介

除了交換網路（如 ATM 網路）是透過交換機來達成工作站之間的連線外，一般區域網路大多是透過共享媒介來通訊，亦即所有工作站都是共同使用同一條傳輸媒介來傳送與接收訊息。因此，網路上必須制定規則，讓工作站依照這個規則來取得媒介的使用權，這也是制定通訊協定的緣由。早期 IEEE 802 制定了三個樣板協定：CSMA/CD、Token-Ring 與 Token-Bus。其中 CSMA/CD（IEEE 802.3）是利用多重存取（Multiple Access）的原則，以競爭的方法，先強奪傳輸媒介的工作站有傳輸資料的權利，但這種協定很容易發生工作站之間資料的碰撞，甚至高負載時也會因碰撞機率增加，而導致網路癱瘓。

然而 Token-Ring 和 Token-Bus 網路並非由競爭原則來取得媒介使用權，而是利用一個稱之為『符記』（Token）（Token 為印地安人首領的旗幟）來表示有使用傳輸媒介的權利。也就是說，網路上存在一個 Token，並在網路上輪流傳遞著，任一工作站取得該 Token，便有使用傳輸媒介的權利，因此，又稱為『服記傳遞』（Token Passing）方式。又網路上只有一個 Token，因此可保證網路上同時只有一個工作站傳遞訊息，如此不但可公平分配傳輸媒介使用權，而且訊息也不會發生碰撞的現象。如果光比傳輸效率，Token Passing 網路無論在哪一方面都優於多重存取網路（如 Ethernet）；但 Token 的管理機制複雜（也就是網路卡製作較不容易），網路架設也較為困難，所以沒有 Ethernet 網路那麼普遍；一般都使用於較嚴謹的應用環境，譬如，工廠自動化系統。因此，我們也需要來探討它的運作原理。

Token-Ring 與 Token-Bus 網路之間最大的區別是在於網路拓樸圖，Token-Ring 是採用環狀網路；而 Token-Bus 是利用匯流排架構。在環狀架構上，Token 只要隨著網路的硬體結構來傳遞即可，但在匯流排架構上要排序 Token 的傳遞順序極不容易，更何況還必須維護有關 Token 遺失或多個 Token 的問題。以下分別介紹這兩種網路的特性；另在此附帶說明，

下一章的 FDDI 網路也是屬於 Token Passing 網路的一種【本章僅簡略說明，如需較完整教材請到作者網站下載】。

## 17-2 Token-Bus 網路簡介

IEEE 802.4 『符記匯流排』( **Token-Bus** ) 網路也是 IEEE 所制定的媒體存取控制( MAC ) 規格，與 IEEE 802.3 一樣，都必須配合 802.2 LLC 連接。Token-Bus 的網路基本架構如圖 17-1 所示，首先我們來介紹它的基本特性：

- (1) **傳輸速率**：4 Mbps。
- (2) **訊框**：IEEE 802.4 Token-bus 訊框，可區分為『資料訊框』和『控制訊框』兩大類。最大訊框為 8191 位元組。
- (3) **傳輸媒介**：光纖、同軸電纜或雙絞線。
- (4) **網路拓樸圖**：匯流排架構，以廣播方式傳遞訊息。
- (5) **傳輸媒介存取技術**：採用『符記傳遞』( **Token Passing** ) 方式，取得 Token 之工作站始可傳送訊息，否則只能接收訊息。並提供四種優先權。
- (6) **公平使用頻寬**：以 Token 傳遞輪流使用傳輸媒介，工作站之間取得媒介的傳輸權利公平分配。
- (7) **高頻寬使用率**：正常情況下，網路上只有一個 Token，也只有一個工作站可以發送訊號，不會發生碰撞情形。於傳輸量較高時，工作站之間輪流使用傳輸媒介，因此具有高頻寬使用率。
- (8) **必須維護邏輯環**：Token 在各工作站之間輪流傳遞構成一個邏輯環，成為網路的控制中心。Token-Bus 採分散式管理方式，取得 Token 之工作站必須負責維護邏輯環。

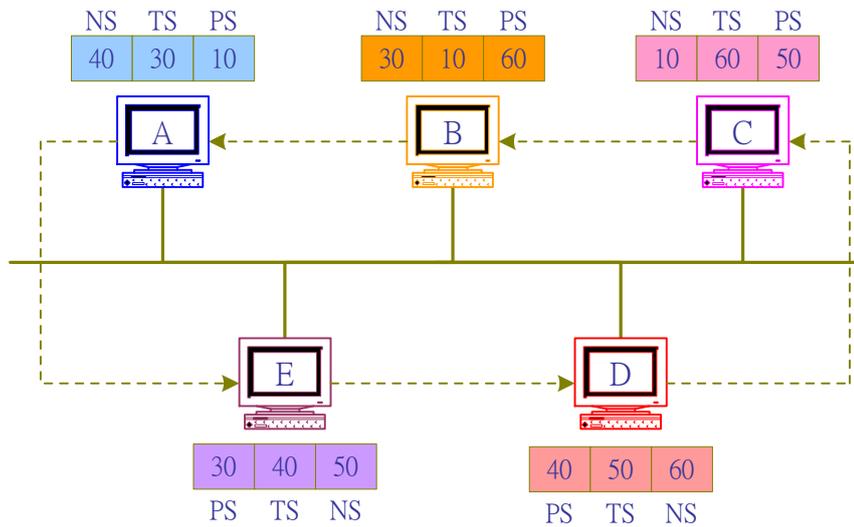


圖 17-1 Token-Bus 網路架構

### 17-3 Token-Bus 運作原理

我們用圖 17-1 和下列幾個重點來介紹 Token-Bus 的運作原理：

Token-Bus 網路的拓樸圖是匯流排架構 ( Bus Architecture )。工作站之間的訊息傳遞和 CSMA/CD 網路相同，都是用廣播方式。任何工作站欲傳送資料給其他工作站，便將訊號廣播到網路上；不是處於傳送資料的工作站，便是在接收情況下，接收到訊息後再判斷是否傳送給本身，如果是，則將資料傳送給 LLC 層，否則將其丟棄。

- (1) 如何協議工作站之間存取傳輸媒介的次序？它是採用『符記傳遞』( Token Passing )的方法。在網路上存在一個『符記』( Token )，任何一部工作站只要取得 Token，便有使用傳輸媒介的權利。沒有取得 Token 的工作站，只能由網路上接收資料。正常情況下，網路上只有一個 Token，也就是只有一部工作站可以傳送資料，因此不會發生碰撞的現象。
- (2) Token 如何在工作站之間傳遞？在網路的運作上有一個『邏輯環』( Logical Ring )，如圖 17-1 所示，由邏輯環上的傳遞順序來決定各個工作站使用傳輸媒介的次序。Token 依照邏輯環的方向傳遞，任何一部工作站取得 Token 就可以傳送資料，當它傳送完後，便將 Token 依照邏輯環傳給下一部工作站。如圖 17-1 中，工作站 A 取得 Token 傳送完資料後，便將 Token 透過網路傳給 E；E 再傳給 D。依此類推，所有工作站便可依照次序使用傳輸媒介。

- (3) 邏輯環應如何建構？為了建構邏輯環，在網路上每部工作站都必須有一個『識別碼』( Identifier, ID )，以標示自己和其它工作站不同(如 MAC 位址)。同時它還要記錄另外二個識別碼：一則是邏輯環上的『前工作站』( Previous Station, PS ); 另一是邏輯環上的『後工作站』( Next Station, NS )。由工作站上 PS 和 NS 的紀錄，就構成了一個邏輯環。如圖 17-1 中，虛線部分為邏輯環，對本身工作站的識別以『本身工作站』( This Station, TS ) 為標示。譬如，工作站 B ( 10 ) 的下一個工作站是 A ( 30 )、再往下一站 E ( 40 ) → D ( 50 ) → C ( 60 ) → B ( 10 ) → A ( 30 )，如此構成一個邏輯環。
- (4) 應如何維護邏輯環？因為邏輯環是完全根據參與工作站中所記錄的識別碼( PS、NS、TS ) 來產生，與硬體的結構無關，也和工作站的實際安裝位置無關，而必須依靠軟體的方法來維護。又 Token-Bus 是屬於分散管理方式，在網路上並沒有一部工作站專門管理邏輯環，而是取得 Token 之工作站必須負責管理。因此，對於邏輯環的維護就比較複雜。
- (5) 邏輯環可能發生哪些異常狀況？邏輯環是由參與運作之工作站輪流管理，可能會因某部工作站運作不正常、或纜線接續不良等因素，造成邏輯環發生異常狀況，譬如，邏輯環故障、符記遺失、或多重符記等等。

## 17-4 Token-Bus 訊框格式

IEEE 802.4 Token-Bus 的訊框可劃分為兩大類：一為傳遞資料時所使用的『資料訊框』；另一為專用來管理邏輯環的『特殊訊框』。以下分別敘述之：

### 17-4-1 資料訊框

一般『資料訊框』如圖 17-2 所示，各欄位功能如下：

- (1) Preamble：建立同步功能並且顯示訊框的第一個位元 (  $\geq 1$  位元組 )。
- (2) SD ( Start Delimiter )：標示訊框的開始 ( 1 位元組 )。
- (3) FC ( Frame Control )：用於表示訊框之功能，以區分本訊框是一般資料訊框或其他特殊控制訊框。FC 欄位的內容如圖 17-3 所示 ( 1 位元組 )。

- (4) DA ( Destination Address ): 本訊框欲傳送的目的地位址 ( 2 或 6 位元組 )。
- (5) SA ( Source Address ): 表示訊框的來源位址 ( 2 或 6 位元組 )。
- (6) LLC-frame : LLC 層所傳遞的訊息 (  $\geq 0$  位元組 )。
- (7) FCS ( Frame Check Sequence ): CRC 檢查碼 ( 4 位元組 )。
- (8) ED ( End Delimiter ): 訊框結束碼 ( 1 位元組 )。
- (9) 最大訊框長度 8191 位元組，從 SD 欄位到 ED 欄位。

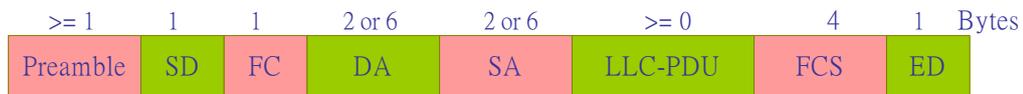


圖 17-2 Token-Bus 訊框結構

## 17-4-2 特殊訊框

『特殊訊框』主要使用於管理邏輯環。工作站取得 Token 後，它便是邏輯環的管理者，必須負責邏輯環的維護工作，以及處理異常狀況的發生。一些常用的特殊訊框如圖 17-3 所示，分別敘述如下：

- (1) 要求—符記 ( Claim Token, CT ): 如果工作站等待太久沒有收到 Token，便送出『要求-符記』訊框。發出訊框等待一個時槽的時間後，若還是沒有收到 Token，則該工作站會繼續發出 CT 訊框，每要求一次會增加一個時槽的等待時間。擁有 Token 者收到 CT 訊框必須立即聲明，否則該工作站會再造出一個 Token。
- (2) 邀請—後工作站—1 ( Solicit-Successor-1, SS1 ): 擁有 Token 者必須負責詢問是否有其它工作站欲加入邏輯環。邀請時發出 SS1 訊框詢問。
- (3) 邀請—後工作站-2 ( Solicit-Successor-2, SS2 ): 此訊框用於邏輯環「可能」發生問題時。當工作站欲將 Token 傳給下一工作站，但對方一直沒反應；而再廣播 W ( Who-Follows ) 訊框後，還是沒有反應。此時，該工作站判定網路已發生嚴重問題，必須重新建構邏輯環，因此會送出 SS2 要求後工作站。
- (4) 誰—跟在後面 ( Who-Follows, W ): 查問誰是本站站的下一個 Token 接受者。

- (5) **解決—爭端 ( Resolve-Contention, R )**：當管理者發出 SS1 或 SS2 訊框後，如果有兩部以上工作站同時反應，而發生爭執時，則該工作站會發出 R 訊框以解決紛爭。
- (6) **符記 ( Token, T )**：符記訊框格式。擁有者即成為當時的網路主控者，只有它才可傳送資料，並負責管理網路。如工作站欲退出邏輯環，也必須等到擁有 Token，並告知它的前站和後站以後才可退出。
- (7) **設定—後工作者 ( Set-Successor, S )**：當管理者送出 SS1、SS2、或 R 訊框後，合乎條件的工作站用 S 訊框來回應欲成為管理者的後工作站。

在圖 17-3 中有一些特殊訊框格式的後面會跟著若干個的『反應時窗』( Response Window )。一個反應時窗是指訊號在匯流排上的『來回傳遞時間』( Round-Trip Propagation Time )，也稱為一個『時槽』( Time Slot )。當 Token 擁有者發出這類的控制訊框時，它會利用不同個反應時窗的時間來監聽匯流排上的訊息，以瞭解其它工作站對這個訊框的反應。在每一個反應時窗中可能發生的反應情況如下：

- (1) **沒有反應**：在這段時間內匯流排上聽不到任何訊息。
- (2) **唯一反應**：聽到一個正確的訊息。
- (3) **多重反應**：在這段時間內，可能有二部以上工作站回應訊息，而導致訊號之間互相干擾 ( 碰撞 )，結果聽到不正確的訊息 ( 也許是雜訊 )。

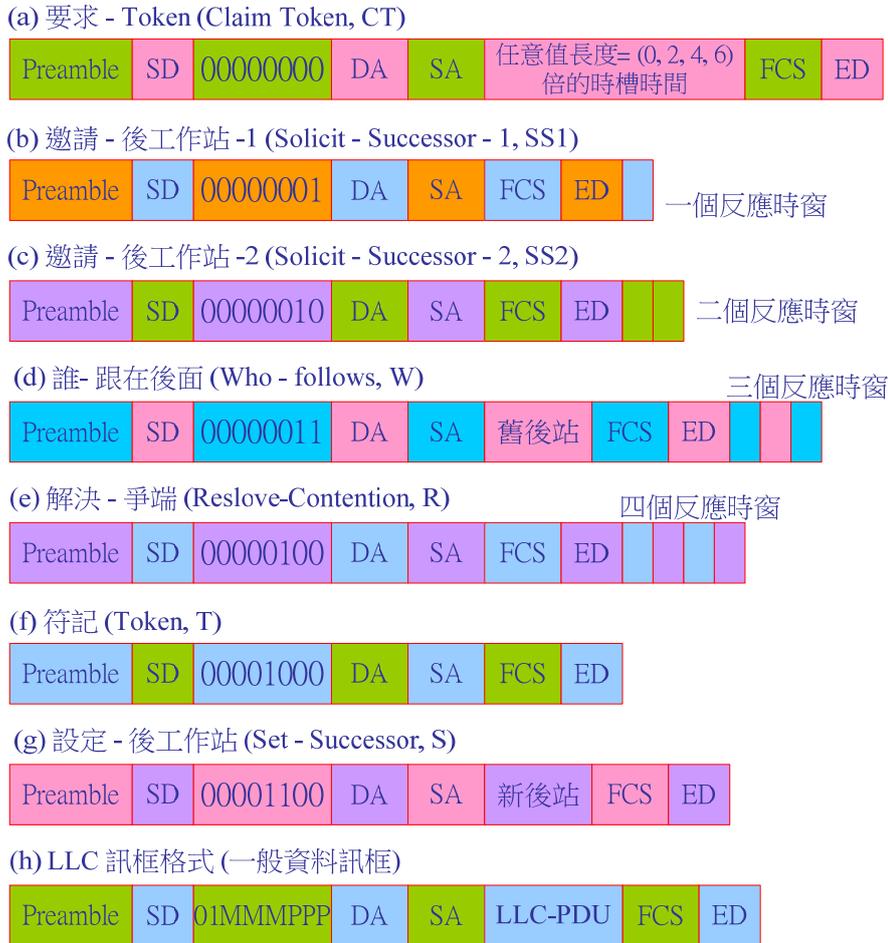


圖 17-3 Token-Bus 特殊訊框格式

## 17-5 邏輯環管理程序

為了使邏輯環的維護比較容易，在正常的運作下，Token 是依照工作站的識別碼，由大到小的順序傳遞。如圖 17-1 中所示，符記的傳遞是依 60 → 50 → 40 → 30 → 10 → 60 的順序。在邏輯環上每一部工作站都有以下的資料：

- (1) TS ( This Station )：記錄本工作站的識別碼，一般皆用 MAC 位址。
- (2) NS ( Next Station )：記錄邏輯環中下一個工作站的識別碼。
- (3) PS ( Previous Station )：記錄邏輯環中上一個工作站的識別碼。

為了避免 Token 被獨占使用，而影響整個網路的傳輸效率，因此工作站上都設有一個『符記保留時限』( Token Holding Timer, THT )，以限制其擁有 Token 的時間，而時限的值是在網路系統建立時就設定的。當資料傳遞完畢或 THT 逾時後，擁有 Token 者就必須將 Token 傳

遞給下一個工作站。而當工作站收到 Token 時，也必須在一個反應時窗（一個時槽）內作出下列反應：

- (1) 傳送資料：當有資料要傳送時。
- (2) 將 Token 傳遞給下一個工作站：當沒有資料要傳送時。

工作站收到 Token 後不可以沒有反應，否則前工作站會認為 Token 已流失或工作站損壞，然後做一些異常狀況的處理程序，造成網路混亂。

邏輯環是否正常運作，直接影響到網路的傳輸效率。工作站取得 Token 時，它就是網路的管理者，必須負責下列正常管理程序：

- (1) 新工作站加入程序
- (2) 工作站退出程序

### 17-5-1 新工作站加入邏輯環

工作站啟動時，並未加入邏輯環內，也無法參與網路的運作，因此，需要一個新工作站加入程序，讓新的工作站加入邏輯環中的適當位置（邏輯環有按識別碼大小順序排列）。持有 Token 者必須負責詢問是否有新工作站欲加入，當它欲將 Token 傳遞出去之前，會先在網路上廣播一個『邀請 - 後工作站 - 1』（SS1）的控制訊框，邀請新的工作站加入，並以一個反應時窗的時間等待新工作站回應。該 SSI 上的控制訊息為：

DA = NS ( 如，20 )

SA = TS ( 如，40 )

DATA = NULL

所有欲加入邏輯環者比較自己的識別碼（如，30）是否介於 DA 和 SA 之間（符合邏輯環大小順序，20、30、40），如果是，就可以在反應時窗內送出一個『設定 - 後工作者』（S）的控制訊息，以表明欲加入邏輯環內。該 S 訊框的訊息為：

DA = 原符記持有者的識別碼（發送 SS1 者）（如，40）

SA = TS ( 如，30 )

DATA = TS ( 如 · 30 )

發送 SS1 的管理者會根據在一個反應時窗中所得到的不同反應而採取適當的動作。它可能有下列三種反應：沒有反應、唯一反應或多重反應，以下分別敘述之：

### (A) 沒有反應

表示沒有介於 DA 和 SA 之間的工作站欲加入邏輯環。因此可以將符記傳給下一個工作站。

### (B) 唯一反應

表示有一個工作站想加入，而且它的識別碼介於 SA 和 DA 之間。我們以圖 17-4 來說明其運作程序。原來邏輯環上有四個工作站，其符記傳遞是依照 20 ( D ) → 60 ( A ) → 50 ( B ) → 40 ( C ) → 20 ( D ) → 60 的順序。而另一個識別碼為 30 的工作站欲加入邏輯環，其加入程序如下：

- (1) 當工作站 40 取得符記並傳送完資料後，它有責任詢問是否有其他工作站想要加入邏輯環。於是它發送『邀請 - 後工作者』( SS1 ) 訊框，其中 DA = 20 ( NS )、SA = 40 ( TS )。
- (2) 這時候工作站 30 在網路上收到 SS1 控制訊框，而且比較自己的識別碼介於 NS 和 TS 之間 (  $20 < 30 < 40$  )，於是發送出『設定 - 後工作站』( S ) 訊框，其中 DA = 40、SA = 30、DATA = 30。由於 S 和 SS1 訊框都適用廣播方式 ( 在匯流排上廣播 )，所以網路上所有工作站都收到這兩個訊框。
- (3) 工作站 40 就將本身記錄的 NS 改為 30 ( NS = 30 )，並將 Token 傳遞給工作站 30。
- (4) 同時工作站 20 也知道工作站 30 已加入邏輯環，而且是自己的前站，因此將 PS 改為 30 ( PS = 30 )。
- (5) 工作站 30 就將 NS 設定為 20、同時把 PS 設定為 40。新的邏輯環便建構成功，符記傳遞順序便更改為 60 → 50 → 40 → 30 → 20 → 60。

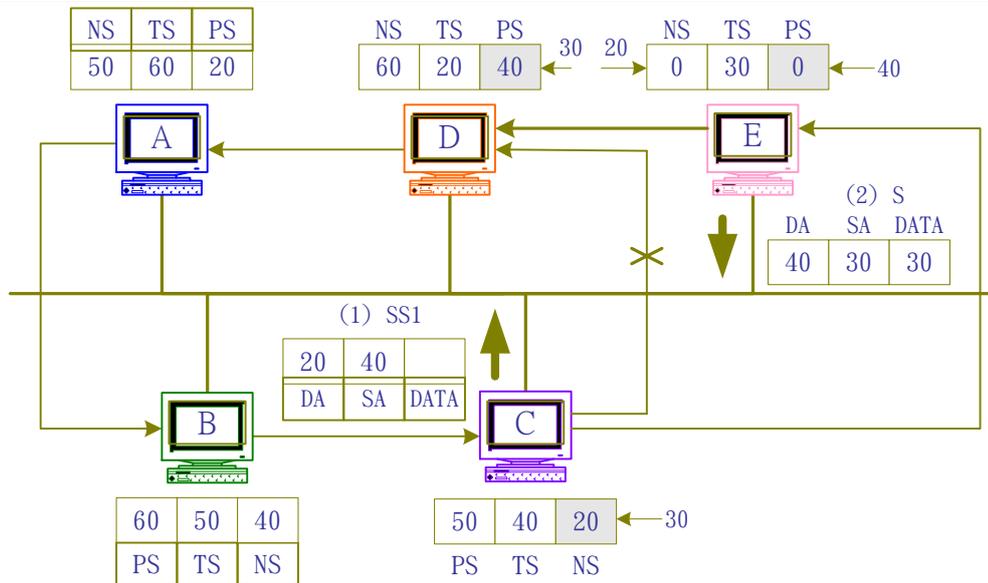


圖 17-4 一個新工作站 (E) 的加入程序

### (C) 多重反應

發生多重反應時，管理者會收到一個被干擾的訊息，表示在反應時窗的時間內，有二個以上工作站發出 S 訊框表示欲加入邏輯環。由於系統只能一次加入一個工作站，因此必須解決爭執。管理者必須再送出『解決 - 爭端』(R) 訊框，在 R 訊框後面緊跟著四個時窗時間，欲加入邏輯環的工作站依照自己識別碼的前兩位元的大小，在時窗時間內送出『設定 - 後工作者』(S) 訊框，其關聯如下圖所示：

11	10	01	00
1	2	3	4

每一個參與競爭者皆依照其識別碼的最左邊二個位元 (最高位元) 來決定是否送出 S 訊框。如前兩位元是 11 便在第一個反應時窗送出、10 在第二時窗送出、01 在第三時窗、00 則在第四時窗。如果在自己的反應時窗以前就聽到網路上有人傳送 S 訊框，表示有識別碼較大者參與競爭，自己便應先退出，等待下次機會再要求加入邏輯環。

如果在較高的反應時窗只有一個工作站反應並送出 S 訊框，管理者就設定它為下一個工作站，並將 Token 傳遞給它，再由新的 Token 持有者負責邀請新的工作站加入。上一次沒有成功加入者，可再依照加入新工作站程序要求加入。

也有可能在同一個較高的反應時窗有二個以上的工作站送出 S 訊框，造成訊號干擾的現象，此時管理者就必須再送出第二個 R 訊框以解決爭端。而上次送出 S 訊框的新工作站也

會從網路偵測到訊號碰撞的現象，知道有其他工作站也同樣符合條件( 識別碼前二位元相同 )，必須再競爭一次。參與競爭者收到第二次 R 後，就再依照最左邊第 3 和第 4 位元的大小，在對應的反應時窗送出 S 訊框。如又碰撞則再比較第 5 和第 6 位元大小，依此類推。我們用一個例子來說明其運作情況，如圖 17-4 中邏輯環的傳遞方向為 60 → 50 → 40 → 30 → 20 → 60。當工作站 50 拿到符記，而它必須負責處理新工作站的加入。當它發出 SS1 訊息時，工作站 45、46、47 都符合條件而要求加入，因此皆送出 S 訊息，也就發生碰撞。工作站 50 連續發出 3 個 R 訊框皆產生碰撞，到了第四個 R 才解決競爭，找出最大的識別碼是 47。完成一次加入程序後，工作站 50 便將符記傳遞給 47( 運作程序如圖 17-5 所示 )，再由工作站 47 去負責解決 45 和 46 兩個新工作站加入的競爭。

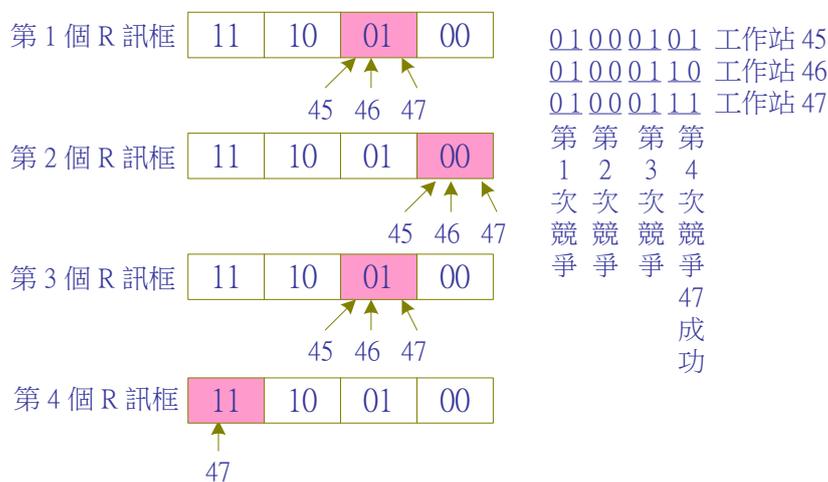


圖 17-5 解決多重工作站加入的競爭

邏輯環上所有工作站 ( 只要擁有 Token ) 都可以發出 SS1 邀請新工作站加入，但只有最小識別碼的工作站不可以。因為當工作站發出 SS1 訊息時，想要加入邏輯環的工作站的識別碼必須符合  $DA < TS < SA$  的條件，也就是說它的識別碼必須介於兩個邏輯環上的端點之間才可以。如果最小識別碼的工作站發出 SS1 訊框，所有網路上介於最大與最小識別碼的工作站都可以參加競爭，因此會破壞網路上由大到小的傳遞順序。所以，最小識別碼者必須送出另一個『邀請 - 後工作站 - 2』( SS2 ) 訊框來邀請新工作站。SS2 訊框後面接著二個反應時窗，第一個是讓小於最小識別碼的工作站反應；而第二個是讓大於最大識別碼的工作站使用。當第一個反應時窗無人使用時，才可以反應在第二個時窗。也就是說，當第一個反應時窗沒有反應，則表示沒有比最小者還小的工作站欲加入。此時，識別碼大於最大識別碼的工作站可以利用第二個反應時窗來反應。欲加入的工作站還是發送『設定 - 後工作站』( S ) 來

告知，如果沒有碰撞現象就依上述 (B) 唯一反應的處理，如發生碰撞就依據 (C) 多重反應的程序來解決紛爭。

### 17-5-2 工作站退出邏輯環

工作站若欲退出邏輯環，就必須先取得 Token 並處理完退出程序後才可以退出。處理程序是送出『設定 - 後工作站』(S) 控制訊息給它的前工作站 (在網路上廣播)，其 S 訊框的內容為：

DA = PS

SA = TS

DATA = NS

而其後工作站收到這個訊息後，會將其 PS 值設定為 S 訊框中的 DA 值；又其前工作站也會收到該訊框，而將其 NS 值設定為 S 控制訊息中的 DATA 值。如此該工作站便在邏輯環中自動消失。此時原 Token 持有者再將 Token 送給新的後工作站。我們用圖 17-6 來說明工作站 30 如何退出邏輯環，當它取得 Token 後(由工作站 40 傳遞過來)，便送出 S 訊框，其中 DA = 40(前工作站)、SA = 30(本工作站)、DATA = 20(後工作站)。後工作站(20)收到該訊框後，將本身的 PS 設定為 40；而前工作站(40)則將本身的 NS 修改為 20。新的邏輯環於是建立成功，工作站 30 也退出邏輯環了。隨後工作站 40 立即再產生一個 Token 傳遞給工作站 20，整個邏輯環又繼續正常運作。

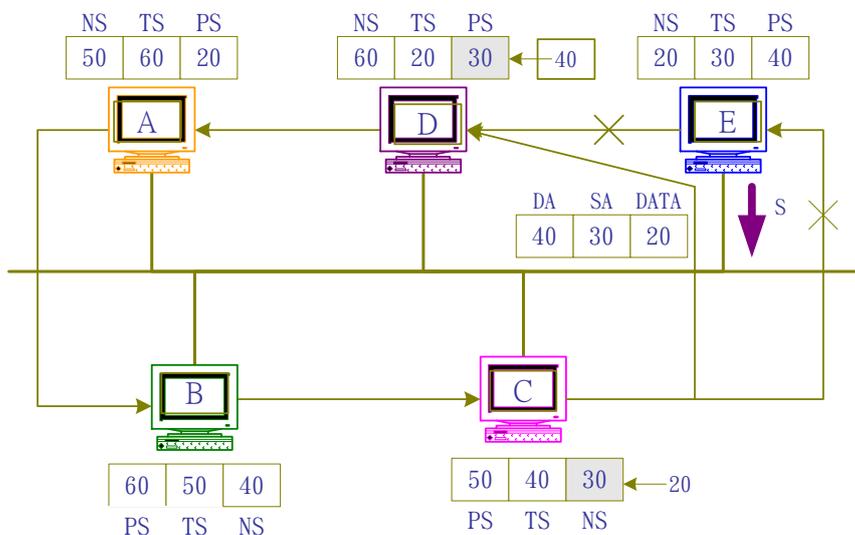


圖 17-6 工作站退出程序

## 7-6 Token-Ring 網路簡介

『符記環狀』(Token-Ring)的標準規範是 IEEE 802.5，也是屬於 IEEE 802 系列中的『媒介存取控制』(MAC)協定，如同 IEEE 802.3 CSMA/CD 協定一樣，它的上層通訊協定必須配合 802.2 LLC 協定。也就是說，對於鏈路的多工處理還是由 LLC 層負責，Token-Ring 協定只提供單一鏈路的媒介存取控制。首先，我們針對 Token-Ring 網路的基本特性歸類如下：

- (1) **傳輸速率**：4Mbps 或 16Mbps。
- (2) **訊框**：IEEE 802.5 Token-Ring 訊框。
- (3) **傳輸媒介**：主要為雙絞線 (STP、UTP) 或光纖。
- (4) **網路拓樸圖**：環狀 (Ring) 網路架構，如圖 17-7 所示。
- (5) **訊號傳輸方式**：點對點 (Point-to-Point) 方式傳送，並容易達成傳送資料之回覆工作。
- (6) **傳輸媒介存取技術**：採用『符記傳遞』(Token Passing) 方式。Token 依環狀網路以單方向傳遞，取得 Token 之工作站方可傳送訊息；否則只能接收訊息，並提供八種優先權。
- (7) **公平使用頻寬**：Token 依照網路硬體所構成之環路傳遞，每個工作站都有公平的機會可取得傳輸媒介使用權。而且高負載時網路線路使用率高。
- (8) **不用維護邏輯環**：由於 Token 傳遞的邏輯環由硬體架設，因此不用維護邏輯環 (不同於 Token-Bus)，但需維護 Token。工作站與網路之間是利用『主動式訊號增益器』(Active Repeater) 連接，網路可連接較長距離，無需另外安裝訊號增益器。
- (9) **適合即時應用系統 (Real-time Application)**：因為工作站持有 Token 的時間有限制，因此，可提供一定期間內的延遲保證，也可預估每一筆資料最遲可送出的時間，達到即時性的要求。

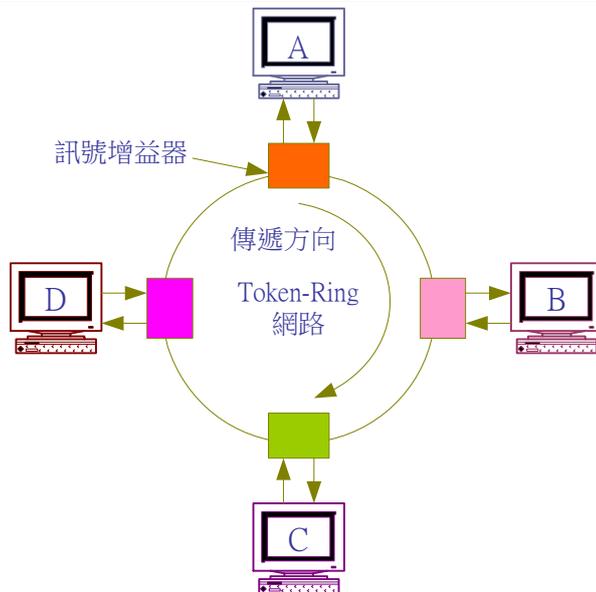


圖 17-7 Token-Ring 網路基本架構圖

## 17-7 Token-Ring 運作原理

我們用圖 17-7 和下列幾個工作重點來描述 Token Ring 網路的運作原理。

- (1) Token-Ring 網路架構為環狀網路，工作站透過主動式訊號增益器接上網路，如圖 17-7 所示。
- (2) 網路上媒介存取方法是符記傳遞方式。也就是說，網路上存在一個符記，取得符記的工作站才可以發送訊息；當它傳送完資料後（或確定不傳送資料），便依照環狀網路傳遞方向將符記傳給下一個工作站，網路上的工作站就這麼依序輪流取得傳輸媒介的使用權，大家公平地來使用媒介。
- (3) 資料訊息在網路上是單一方向傳遞。沒有正在傳送訊息的工作站，便處於接收狀態。當訊息傳遞到工作站，位元串列依序進入工作站，每一位元經過一位元時間的延遲後，再由該工作站送出。接收工作站將全部訊框皆複製後，再判斷它傳送的目標是否為本工作站，是則往上一層 LLC 傳送，否則將其拋棄。
- (4) 訊息資料由傳送端發出，在正常情況下必定回到傳送端（環狀網路）。傳送端必須負責將其發送的訊框收回。

由以上的敘述，我們大略可以瞭解 Token-Ring 網路的特性，它也是利用 Token 傳遞來決定傳輸媒介的使用權。正常情況下，網路上只有一個 Token，因此不會發生碰撞現象。我

們可以回顧 Token-Bus，它也是利用 Token 傳遞方式來解決媒介使用權的紛爭，但每一部工作站必須花費許多額外工作（Overhead）的時間來維護邏輯環，對整個網路而言，不僅管理上過於複雜，也易造成網路不穩定（干擾訊號破壞符記的傳遞）。Token-Ring 的做法則乾脆將邏輯環建造成實體連線，因此不需要維護邏輯環，也可減輕工作站的負荷，網路維護亦較簡單。最重要的是網路上不會發生碰撞，又平均使用傳輸媒介，在高負載即時傳送的环境裡，是很好的通訊協定。

在 Token-Ring 網路上的 Token 可依當時的工作情況分為兩種：

- (1) 『自由符記』( Free Token ): 利用某一個位元串 ( 如 1111111 ) 或某一個控制位元 ( 如 AC 中的 T=1 位元 ) 來代表自由符記，又稱為 『符記』，只代表權力的交替，並不攜帶資料。
- (2) 『忙碌符記』( Busy Token ): 利用某一個位元串 ( 如 1111110 ) 或某一個控制位元 ( 如 AC 中的 T=0 位元 )，來代表忙碌符記，又稱為 『訊框』( Frame )，表示其攜帶著訊息在傳送。

Token 的傳遞運作如下：網路上任何一個工作站取得 Free Token ( 或稱符記、Token )，便立即將 Free Token 更改為 Busy Token，並將欲傳送的資料緊跟著 Busy Token 之後 ( 構成資料訊框、Frame ) 傳送出去。傳送完並收回訊框後 ( 訊框循環狀網路繞回原點 )，立即將 Busy Token 更改為 Free Token，同時傳送給下一個工作站。

如圖 17-8 (a)，工作站 A 欲將資料傳送給工作站 C，當它取得 Free Token ( T = 1 ) 後，便立即將 Free Token 改為 Busy Token ( T = 0 )，並且連帶資料一起傳送出去 ( 稱為訊框 )，如圖 17-8 (b)。往前送的資料位元串每經過一個工作站時，都會先被複製一份後再往下一站傳送，因此整個資料串列每經過一個工作站就產生一位元時間的延遲。當整個訊框都經過時，每部工作站都將訊框複製一份。工作站 C 判斷該訊框是傳送給自己，便將其保留，其餘工作站則將其拋棄。該訊框經過環狀網路環繞後會回到工作站 A，如圖 17-8 (c)。隨後工作站 A 將訊框收回，再將 Busy Token ( T = 0 ) 改為 Free Token ( T = 1 )，並循環路方向傳給下一個工作站 ( B )，如圖 17-8 (d)，再由工作站 B 決定是否傳送資料。

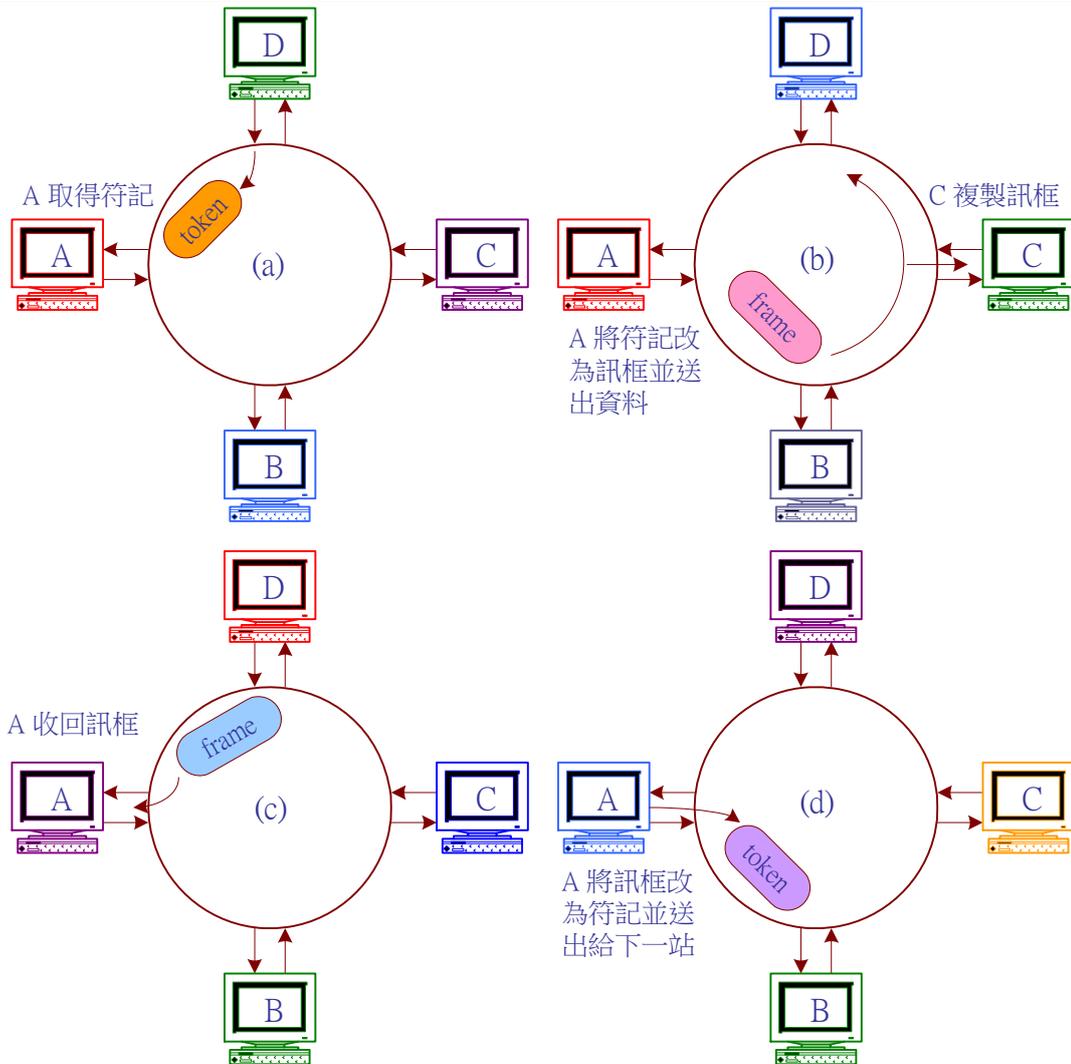


圖 17-8 Token-Ring 網路的運作

## 17-8 訊號增益器

Token-Ring 網路是利用訊號增益器連接工作站和環狀網路，我們分別介紹其功能和狀態：

### 17-8-1 訊號增益器之功能

訊號增益器的主要功能有：

- (1) 將訊框加入環狀網路中 ( **Frame Insertion** )：每一工作站取得 Token 後，便可透過訊號增益器將訊框發送至網路上。

- (2) **接收網路上訊框 ( Frame Reception )** : 未持有 Token 者負責傳遞資料，訊框依序進入訊號增益器，將其複製一份後再往前發送。當整個訊框皆複製後再判斷其目的位址 ( Destination Address ) 是否為自己，是則保留；否則丟棄該訊框。
- (3) **移除訊框 ( Frame Removal )** : 由於網路結構是環狀，如果發送的訊框沒有移除的話，它將會不停的在網路上循環繞行，所以必須由發送端的訊號增益器將訊框移除。

## 17-8-2 訊號增益器之狀態

訊號增益的工作狀態有下列三種：( 如圖 17-9 所示 )

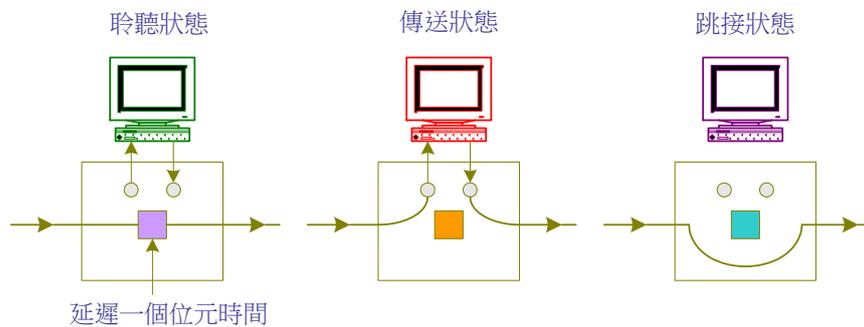


圖 17-9 訊號增益器狀態

- (1) **聆聽狀態 ( Listen State )** : 工作站在接收情況時，訊號增益器便處於聆聽狀態。資料串的每個位元依序進入訊號增益器，並經過一個位元時間的延遲後再轉送出去。利用這一位元時間，可同時完成下列工作：
- (a) 巡視經過的位元串 ( Bit Stream )，並做特殊位元串的辨識工作，如辨識 Token 或目的地位址等等。
  - (b) 將進入的位元複製一份給工作站。
  - (c) 在一個位元通過時修改其值。如將 Free Token (  $T = 1$  ) 修改成 Busy Token (  $T = 0$  )，或加入確認位元等等。
- (2) **傳送狀態 ( Transmit State )** : 工作站在傳送資料的狀態下，訊號增益器將訊息傳送出去，並且由另外一端收回訊息。

(3) **跳接狀態 ( Bypass State )**：如果工作站故障或關機時，訊號增益器會自動進入跳接狀態，訊號可直接經過而不作任何處理。

## 17-9 Token-Ring 網路架構

如果依照環狀網路基本架構( 如圖 17-7 )來佈線，Token-Ring 網路維護上將非常的困難。試想我們由起始點開始佈線，而最後這一條線必須回到起始點，如果經過的區域很大，佈線成本將非常的高。又若網路上任何一個區域斷線，將使整個網路癱瘓，因此維護上十分困難。所以 Token-Ring 也採用類似集線器 ( Hub ) 的佈線方法，將環狀網路轉換成星狀的拓樸圖，如此不但佈線容易，維護也較為簡單。

Token-Ring 網路是利用一個稱為『環路集中器』( Ring Wiring Concentrator, RWC ) 的裝置來當成佈線中心，每個集中器上有多個『線接埠』( Wiring Lobe ) 用來連接工作站 ( 一般都有 12 個埠口 )，再由若干個集中器構成網路傳輸骨幹。集中器內部的交換線路有『線接埠跳接』( lobe-bypass ) 的功能，可以將受損工作站 ( 或連線斷線 ) 的接點『跳接』( bypass )，使其排除於網路之外。另外它也具有網路維護及重新組織的功能。我們可以串接若干個集中器來構成星狀網路，這樣的佈線在維護和安裝上比較容易，如圖 17-10 所示。

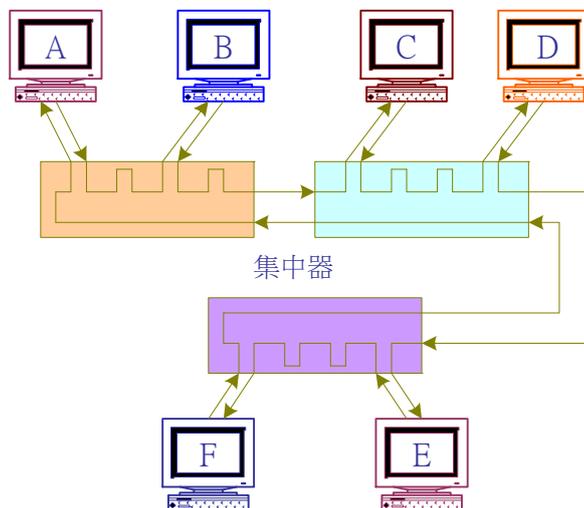


圖 17-10 運用環路集中器的 Token-Ring 網路佈線圖

### 17-9-1 網路硬體故障處理

網路硬體故障是表示網路線路斷線、或是某個工作站的傳送器或接收器發生故障，造成網路的環路斷線，訊號無法透過傳輸線順利繞行。一般 Token-Ring 網路的佈線皆採用『集

中器』(RWC) 為網路主要幹線，所有工作站透過集中器的『線接埠』(Wiring Lobe) 連接，集中器上的連線是屬於雙環網路：包括一對『主環』(Primary Ring) 和『備用環』(Backup Ring)。兩個環路方向相反，如圖 17-11 所示，正常傳送資料是透過主環傳送，而備用環是當主環路斷線或集中器故障時，做為環路跨接使用。

如果集中器上的工作站發生故障（也許是傳送器或接收器故障），導致下一個工作站一直接收不到訊號，此時該收不到訊號的工作站會送出『求救』(Beacon) 控制訊框，其中記載著自己和前工作站的位址。這個求救訊框沿著主環往前送，所經過的工作站必須停下工作，並立即讓求救訊框經過。如果集中器上某一個線接器送出訊號後無法再收回，表示該線接器上的工作站發生故障，此時就必須將該線接器跳接。如圖 17-11 中，工作站 80 一直沒有收到任何訊息，便發送『求救訊框』，工作站 90、70、40、60 收到求救訊框後，立即讓它通過；但該訊框無法到達工作站 20，因此它所連接的『線接埠』就被跳接 (Bypass)。

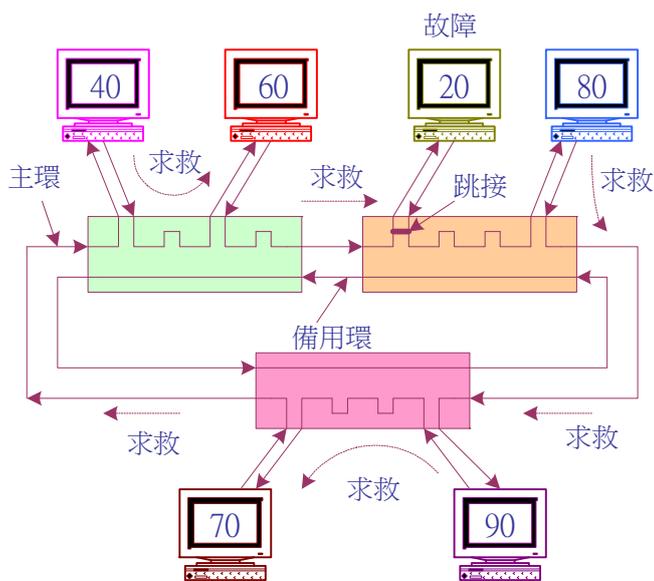


圖 17-11 故障工作站跳接程序

如果線路斷線或集中器故障，將會使整個網路停頓，這種情況的處理程序就更為重要，而且會用到備用環。同樣的，當某部工作站發現無法傳送資料給後工作站，便發出求救訊框（記載本工作站位址和前工作站位址）。而集中器若無法讓求救訊框通過，就會跳接到備用環路，重新構成一個迴路。如圖 17-12 中，當工作站 30 發現無法將訊號傳送出去，便發送求救訊框；但求救訊框無法經過集中器 3 傳送出去，這時候便跳接到備用環。求救訊框繼續沿著備用環傳送，到集中器 1 以後又無法通過，集中器 1 便將備用環跳接到主環，求救訊框再沿著主環傳送回到工作站 30。至此斷線的地方已被隔離開，且主環和備用環亦順利構成新

的環狀網路。如集中器故障也是一樣，故障點被隔離，其他工作站依然正常運作；但故障集中器所連接的工作站就無法正常運作了。

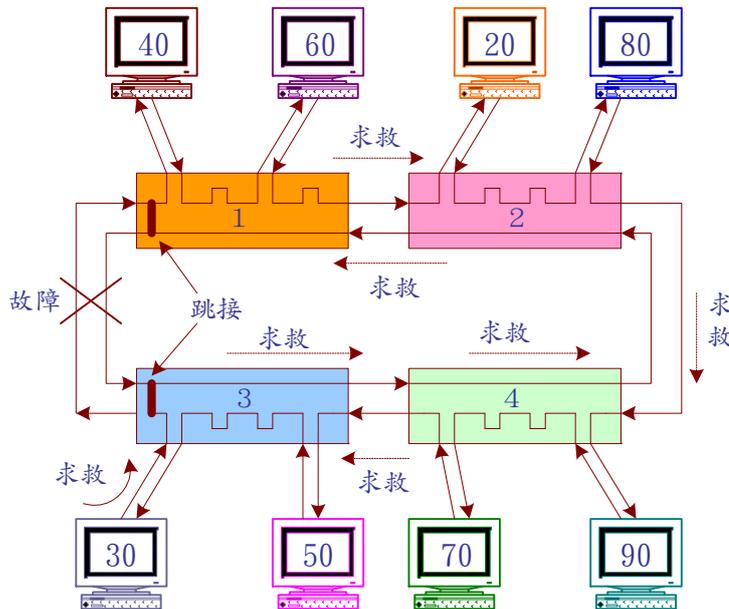


圖 17-12 線路故障雙環網路重新組織程序

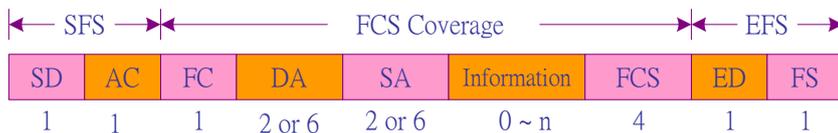
## 17-10 Token-Ring 訊框格式

圖 17-13 為 Token-Ring 的訊框格式，原則上訊框格式可分為兩大類：『資料訊框』和『符記訊框』。如同其它 MAC 協定一樣，定址也是允許使用 2 或 6 個位元組，但在同一網路上只能使用一種方式。以下分別介紹各欄位的功能：

(a) 符記訊框格式



(b) 資料訊框格式



(c) DA/SA 位址格式

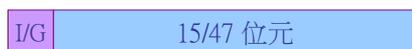


圖 17-13 IEEE 802.5 Token-Ring 訊框格式

- (1) **起始符號 ( Starting Delimiter, SD )**: 此欄位為訊框的起始符號，它的組成是 JK0JK000 位元串，使用非資料編碼，所以不會和資料混淆。
- (2) **存取控制 ( Access Control, AC )**: 資料和符記訊框都會包含此欄位，其格式為 PPPTMRRR。前三個位元( PPP )用來表示本訊框的優先等級( 0 ~ 7 等級 )。T( Token ) 位元用來分辨本訊框是符記 ( T = 1 ) 或資料 ( T = 0 )。M ( Monitor ) 位元用來表示本訊框是 ( M = 1 ) 否 ( M = 0 ) 為『監督站』。預約位元 ( RRR ) 是用來讓其他工作站預約想要傳送訊框的優先權，它只會存留優先權最高的。
- (3) **訊框控制( Frame Control, FC )**: 用於辨識本訊框是傳送 MAC 控制訊息或 LLC-PDU，其格式為 FFCCCCC。前兩個位元( FF )是用來分辨所傳送的是 LLC-PDU 或 MAC 控制訊息。最後六個位元 ( CCCCC ) 是當該訊框作 MAC 管理時，存放訊框狀態或網路管理訊息之用。
- (4) **位址欄位( Address Field, AF )**: 包含來源位址( SA )和目的位址( DA )各 2 或 6 bytes 長度。前一個位元 ( I/G ): I 表示獨立個體位址；G 表示後面的位址是群體位址，如圖 17-13 (c) 所示。
- (5) **訊息 ( Information )**: 內容為 LLC-PDU 或 MAC 控制訊息。
- (6) **訊框檢查碼 ( Frame Check Sequence, FCS )**: 檢查範圍是由訊框控制欄位 ( FC ) 到訊息欄位之間的循環多餘碼檢查 ( CRC-32 )。
- (7) **結束符號 ( End Delimiter )**: 表示訊框結束的位置。如同起始符號一樣，利用非資料編碼方式來標示 ( J、K )。它的組成是 JK1JK10E 的位元字串，其中 E 位元是『軟體錯誤指示』( Soft Error Indicator, SEI )，此位元若被設定，即表示已受環境的影響收到錯誤的資料 ( 如干擾等 )，而非因硬體故障而導致錯誤。
- (8) **訊框狀態( Frame Status )**: 此欄位是讓接收工作站回覆確認訊息使用，它的組成是 AC ××AC××。如果接收端已辨識到此訊框的目的位址，它就將『位址已辨識位元』( Address Recognized Indicator, ARI )設定為 1 ( A = 1 )。如果接收端已完成接收此訊框，就將『訊框已接收位元』( Frame Copied Indicator, FCI )設定為 1 ( C = 1 )。

因為符記訊框並沒有攜帶資料，本身訊框長度只有三個欄位（24 個位元）：起始符號、存取控制和結束符號欄位。又存取控制欄位中的 T 位元為 1（ $T=1$ ），表示符記。當訊框某一工作站接收到符記訊框後，立即將 T 設定為 0，並將資料緊接於存取控制欄位之後送出，該訊框就直接由 Free Token 一變而為資料訊框（Frame）。

## 17-11 Token-Ring 優先權管理

一般 Token-Ring 網路大多是被用在『即時系統』（Real-time System）的環境裡，主要原因是 Token-Ring 網路具有良好的優先權管理功能。換句話說，在具有優先權的網路系統之下，傳輸媒介的使用權是按照各工作站的訊框優先次序排序，擁有較高優先權訊框的工作站優先取得使用的權利。但各工作站之間誰也不知道他人所欲傳送訊框的優先等級。因此，必須有一個優先權管理程序來負責協調處理。

我們用圖 17-14 來說明即時性網路的運作情形。假設網路上各個工作站都是屬於即時系統（Real-time System），對每一個程式（Process）都附有優先等級。在 LLC 的各個服務存取點（SAP）連結其相對應的程式，應該選擇哪一個服務存取點來使用，邏輯鏈路就依照優先等級排序（Scheduling）。LLC 將 LLC-PDU 的優先權分為 8 個等級，等級 7 為最高優先權，等級 0 為最低。最高優先權的 LLC-PDU 優先取得 MAC 的使用權。

如圖 17-14 中，工作站 A 的最高優先權訊框是 a1（7）、工作站 B 是 b1（4）、工作站 C 是 c1（6）、而工作站 D 是 d1（4），因此在網路上有優先權分別是 7（a1）、4（b1）、6（c1）、4（d1）的四個訊框準備要傳送，如果該網路為即時性網路的話，應該是 a1 優先取得網路使用權，這就是優先權管理原則。

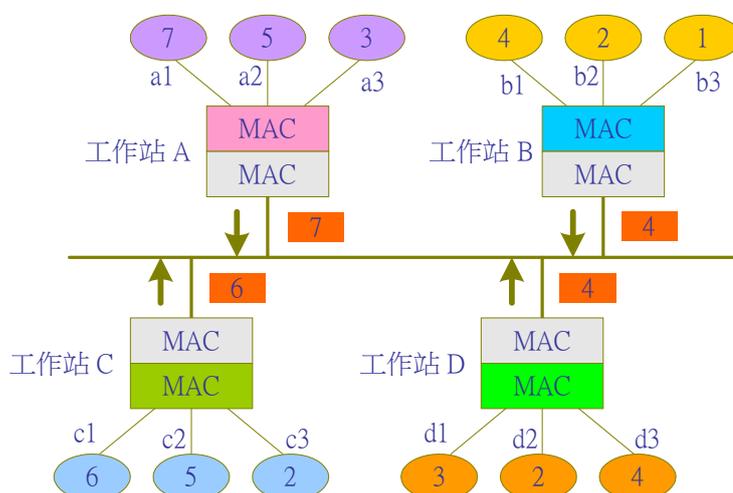


圖 17-14 即時性網路系統範例

在沒有優先權考量的網路上，任何取得 Free Token 的工作站便有傳送資料的權利。但在有優先權的網路環境下，取得自由符記後必須判斷所欲傳送資料的優先等級是否高於其他工作站所欲傳送的，如果自己的資料是網路上所有預備傳送資料之中具最高優先權者，便可立即傳送，否則必須讓給別人。為了做好優先權的處理，不論在 Free Token 或 Busy Token（即資料訊框）上都有兩個欄位 PPP 和 RRR（存取控制欄位，AC），PPP 表示該訊框（符記或資料）的優先等級，RRR 是預約優先等級，主要是讓工作站預約有哪一等級的資料準備傳送。一個 Free Token 在網路上繞行，當它經過每一個工作站時，工作站比較該 Free Token 的優先等級，如果是等於或小於欲傳送資料的等級，這時該工作站就將 Free Token 更改為 Busy Token，也立即將資料送出。當 Busy Token 在網路上繞行，經過任何一個工作站時，工作站會比較該訊框的 RRR 預約等級，如果預約等級小於本身欲傳送資料的等級，便將其更改為本身資料的等級（完成預約動作）。當 Free Token 在繞行時，其所經工作站的優先權如果小於符記上的，雖然不可以取用 Token，但也可完成預約的動作（登錄在 RRR 上）。

當傳送資料者收回 Busy Token 時，該訊框已在網路上環繞一圈，在訊框內的 RRR 欄位上所登錄的一定是目前其他工作站所欲傳送資料的最高優先權，此時該工作站必須將原來的優先權（PPP 的紀錄）儲存下來，且更改符記優先權為預約優先權（RRR → PPP），再將 Busy Token 改為 Free Token 並送出。Token 的優先權不可以一再的上升，否則優先等級較低的資料就沒有機會傳送，因此，每一個有修改過符記優先等級的工作站，必須記錄原來符記的等級（舊的）和修改後的優先等級（新的）。當一個自由符記經過時，如果符記上的等級和紀錄中『新的』等級相同，即表示那一個等級是自己修改的，必須用紀錄上『舊的』等級將其修改回來。也就是說，符記的優先權是誰修改的，誰就必須負責將其還原。

我們用圖 17-15 為例子來說明優先權的處理程序。假設網路上有 A、B、C、D 四個工作站。每一工作站上都有兩個紀錄，New 表示該工作站修改符記優先權時新設定的值，Old 表示修改前符記的優先等級。圖中 "T" 表示自由符記；"F" 表示忙碌符記（資料訊框）。矩形圖形表示傳送的訊框，前者為符記優先等級（PPP）；後者為預約等級（RRR），表示如下：



圖 17-14-1

其運作過程如下：

- (1) 假設由 A 開始，A 傳送資料訊框，該訊框的優先權為 4，預約等級為 0。經過 B 時，B 上有一個等級為 5 的資料預備傳送，因此在訊框上預約登錄 5。經過 D 時，D 上有個資料等級為 6，在比較訊框上的預約登錄後發現比自己的小，便將訊框上的預約登錄為自己的 (6)。訊框回到 A，由 A 收回。
- (2) A 將資料訊框改為符記，將符記的優先權設定為預約等級的值，並將符記傳出。符記經過 B 時，B 比較符記優先權比自己的高，不可取用符記，只好再預約登錄。D 發現符記優先權和自己的相同，就將符記改為資料訊框並傳送資料。
- (3) 資料訊框經過 C 時，被 C 預約登錄為 7。D 收回資料訊框。
- (4) D 將資料訊框改為符記，設定符記優先權為預約登錄的紀錄 (7)，記錄下變更的值 (原優先權 = 6、新優先權 = 7)，並將符記送出。B 再預約 5。C 取得符記並傳送資料。
- (5) C 收回資料訊框並將其改為符記送出。符記到達 D 時，D 發現符記的優先權和自己紀錄上的新優先權(7)相同，便知道這是自己修改的，就用紀錄上的舊優先權(6)替代符記的優先權。
- (6) 符記到達 A 時，A 也比較發現符記上優先權是自己修改的。本來必須將符記優先權設為舊優先權 (4)，但發現預約優先等級 (5) 高於符記本身，便將其優先權設定為預約等級 (5)。符記到達 B，B 取得符記並傳送資料訊框。

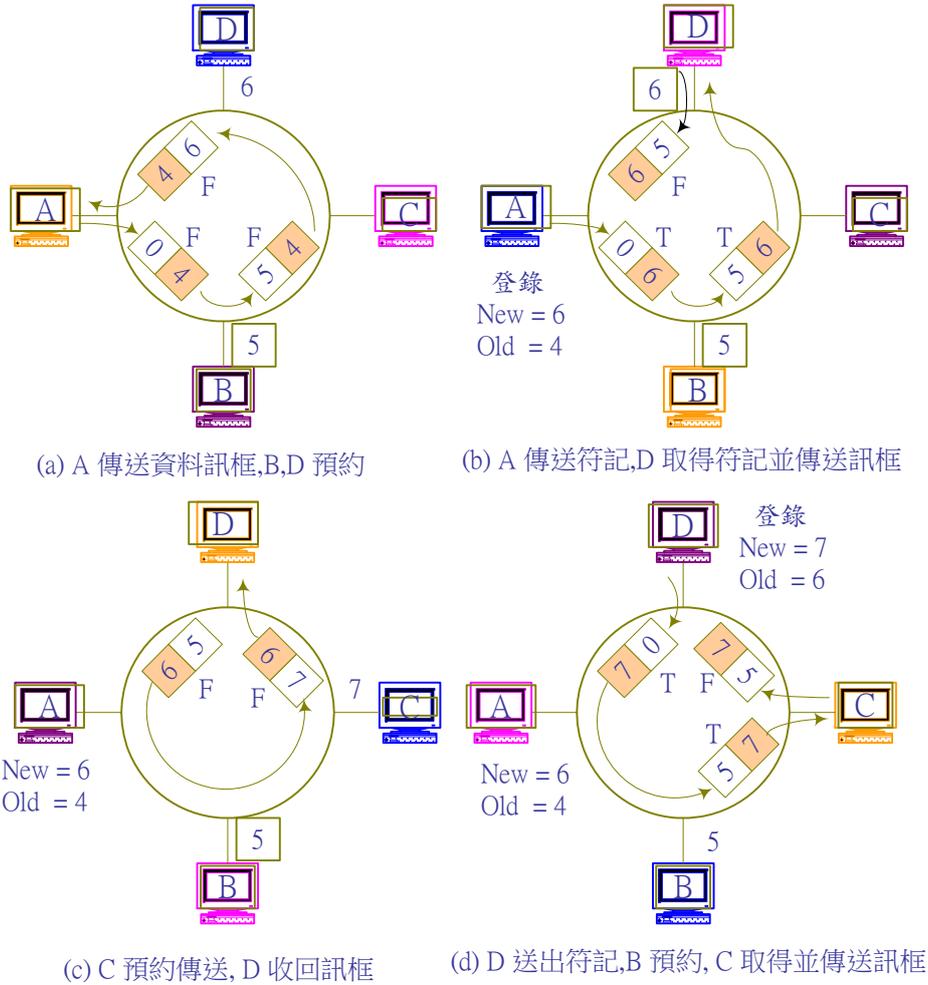


圖 17-15 Token-Ring 優先權處理程序範例

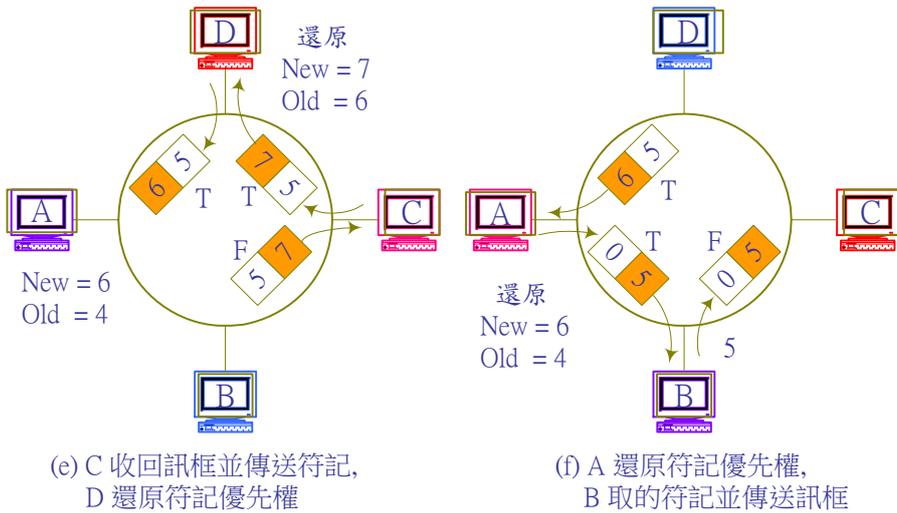


圖 17-15 ( 續 ) Token-Ring 優先權處理程序範例

習題

1. 請繪圖說明 Token-Bus 的運作原理。
2. Token-Bus 網路之中邏輯環和工作站位置是否有連帶關係？並請說明其原因。
3. 請說明為何 Token-Bus 不會發生資料訊框碰撞？為何頻寬可以公平分配？
4. 為何『要求 – 符記』( Claim Token, CT ) 的資料欄位內 ( Data ) 有 0、2、4、6 不同的時槽長度？其功能為何？該欄位內應填寫何值？
5. 有一些 Token-Bus 控制訊框的後面都有接著若干個反應時窗，為何一個反應時窗的長度剛好是一個時槽？而一個時槽的長度如何評估出來？
6. 在圖 17-16 中並沒有邏輯環，網路欲起始建立邏輯環，請簡略說明整個邏輯環建構的過程。

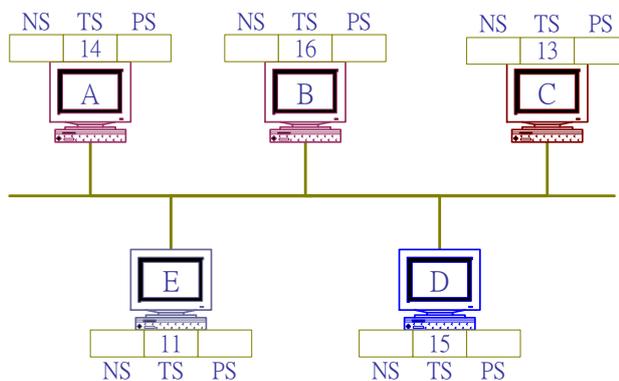


圖 17-16 Token-Ring 範例

7. 如上題，工作站 A 欲退出邏輯環，它的運作程序如何？請說明之。
8. 如上題，工作站 A 退出後，欲再加入邏輯環，請問它的運作程序如何？
9. 為何 IEEE 802.4 資料訊框中沒有長度欄位 ( Length ) ？接收端如何來判斷一個訊框的開始和結束。
10. 請簡略說明 Token-Ring 網路上符記運作的情形？
11. 為何 Token-Ring 網路不會發生碰撞現象？為何頻寬可以公平分配？
12. 為何 Token-Ring 網路適合即時應用環境？為何工廠自動化系統的環境裡都採用 Token-Ring 網路，而不採用 Ethernet 網路？
13. 請敘述 Token-Ring 網路上訊號增益器的功能。

14. 為何 Token-Ring 網路容易達成接收確認的回應？
15. Token-Ring 網路的訊號增益器有哪幾種狀態？各種狀態的功能是什麼？
16. 為何資料串列訊號每經過訊號增益器時就會發生一位元時間的延遲？它的功能是什麼？
17. 在 Token-Ring 的訊框中並沒有長度欄位( Length )，接收端如何判斷訊框的起始和結束？它們的訊號是什麼？
18. Token-Ring 網路上的主環和備用環是否可以佈放在同一電纜上？請舉例說明其原因。
19. 請舉例說明 Token-Ring 的優先權管理。