

發明專利說明書

(本說明書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫)

※申請案號：95146174

※申請日期：95.12.8

※IPC 分類：G06F3/06

0000.01

一、發明名稱：(中文/英文)

具有命令傳遞機制之媒體卡

MEDIA CARD WITH COMMAND PASS THROUGH MECHANISM

二、申請人：(共 1 人)

姓名或名稱：(中文/英文)

美商桑迪士克股份有限公司

SANDISK CORPORATION

代表人：(中文/英文)

1. 查爾斯 樊 歐登

VAN ORDEN, CHARLES

2. 梅根 康普特

COMPORT, MEGAN

住居所或營業所地址：(中文/英文)

美國加州謬佩塔斯市麥卡錫大道 601 號

601 MCCARTHY BOULEVARD, MILPITAS, CA 95035, U. S. A.

國 籍：(中文/英文)

美國 U.S.A.

三、發明人：(共 6 人)

姓 名：(中文/英文)

1. 羅伯特 C 張
CHANG, ROBERT C.
2. 亨利 利卡多 哈頓
HUTTON, HENRY RICARDO
3. 法西 沙貝特-沙爾西
SABET-SHARGHI, FARSHI
4. 哈路特 肯特 泰尼克
TANIK, HALUT KENT
5. 羅 巴利列
BARZILAI, RON
6. 美托 丹 阿里
ARI, MEYTAL DAM

國 籍：(中文/英文)

1. 美國 U.S.A.
2. 美國 U.S.A.
3. 美國 U.S.A.
4. 土耳其 TURKEY
5. 以色列 ISRAEL
6. 以色列 ISRAEL

四、聲明事項：

☐ 主張專利法第二十二條第二項 ☐ 第一款或 ☐ 第二款規定之事實，其事實發生日期為： 年 月 日。

☒ 申請前已向下列國家（地區）申請專利：

【格式請依：受理國家（地區）、申請日、申請案號 順序注記】

☒ 有主張專利法第二十七條第一項國際優先權：

1. 美國；2005 年 12 月 08 日；11/298,349

2. 美國；2005 年 12 月 08 日；11/299,186

☐ 無主張專利法第二十七條第一項國際優先權：

1.

2.

☐ 主張專利法第二十九條第一項國內優先權：

【格式請依：申請日、申請案號 順序注記】

☐ 主張專利法第三十條生物材料：

☐ 須寄存生物材料者：

國內生物材料 【格式請依：寄存機構、日期、號碼 順序注記】

國外生物材料 【格式請依：寄存國家、機構、日期、號碼 順序注記】

☐ 不須寄存生物材料者：

所屬技術領域中具有通常知識者易於獲得時，不須寄存。

九、發明說明：

【發明所屬之技術領域】

本發明大體而言係關於抽取式電子電路卡之使用及結構，且更具體言之，本發明係關於允許個人電腦或其他主機藉由不支援特定媒體卡命令之讀取器及/或主機軟體來使用此等命令。

【先前技術】

近來，諸如緊密快閃卡(Compact Flash Card)、安全數位(SD)卡、多媒體卡(MMC)、xD及索尼記憶棒/高速記憶棒(Memory Stick Pro)之小型快閃記憶卡已得到消費者的廣泛接受。此等裝置主要經設計用於諸如數位相機及快閃記憶體音樂播放器之消費者電子裝置。然而，需要其亦具有至個人電腦之方便連接以供上載及下載資料。

不同卡具有不同電氣介面且通常具有可由用於卡之主機使用的專用於媒體之命令。另外，由於此等卡經啟動以過載有卡與卡之間不同之額外功能，所以卡協定可不僅在卡形式因數之間不同，而且在相同形式因數之各種卡之間亦不同。該等命令通常具有諸如特定輸入-輸出(I/O)及安全操作之獨特功能。因為此等卡之使用變得更加多樣且其不斷用於更多類型之應用程式，所以此等新應用程式通常將涉及現有協定中所缺乏之功能或命令。

具有不同機械及/或電氣介面之市售非揮發性記憶卡之實例包括可自 Sunnyvale, California 之 Corporation of Sunnyvale(本申請案之受讓人)購得的相關多媒體卡

("MMC")及安全數位("SD")記憶卡。存在著符合國際標準化組織("ISO")及國際電工技術委員會("IEC")之標準的其他卡，經廣泛實施之實例係已知為ISO/IEC 7816標準。

在由Cupertino, California之多媒體卡協會(MultiMediaCard Association)("MMCA")不斷更新及公開的"規格 The MultiMediaCard System Specification"中給出MMC之物理及電氣規格。日期分別為1999年6月、2000年1月、2001年6月及2004年2月之該規格之版本2.11、2.2、3.1及4.0以此引用的方式明確地併入本文中。在單一卡中具有達128兆位元組的變化之儲存容量的MMC產品當前可自SanDisk公司購得。在由SanDisk公司公開的日期為2000年4月"MultiMediaCard Product Manual"修訂版2中描述了此等產品，其手冊以此引用的方式明確併入本文中。美國專利第6,279,144號及在1998年11月4日申請之申請案第09/186,064號中亦描述了MMC產品之電操作的某些態樣。在美國專利第6,040,622號中描述了實體卡結構及製造其之方法。讓渡給SanDisk Corporation之此等申請案及專利以此引用的方式明確併入本文中。

較新之SD卡類似於MMC卡，除用以容納額外記憶體晶片的增加之厚度之外，其具有相同大小。其之間的主要不同之處在於：SD卡包括額外資料接點以使得能夠在卡與主機之間更快地傳送資料。(具有額外資料接點之MMC卡之版本亦為可用的，見上文闡述之MMC規格之4.0版)。SD卡之其他接點與MMC卡之彼等接點相同，從而使得經設計以接受

SD卡之插槽亦將接受MMC卡。如以此引用方式併入本文中之美國專利第6,820,148號中所描述的，以經設計以接受SD卡之插槽亦可經製造以接受MMC卡的方式進一步製造與SD卡之間的電氣介面及功能介面。在2000年8月17日申請之美國專利申請案第09/641,023號中描述SD卡之某些態樣，該申請案以此引用之方式併入本文中。(SD卡之規格可用於SD協會(SDA)之成員公司)。

根據ISO/IEC 7816標準而製造之卡具有不同形狀，其具有在不同位置中之表面接點，及一不同於MMC及SD卡之電氣介面。ISO/IEC 7816標準具有總標題 "Identification cards-Integrated Circuit(s) Cards with Contacts"，且由具有自1994年至2000年之個別日期的第1部分至第10部分組成。此標準(其複本可自Geneva, Switzerland, 之ISO/IEC獲得)以引用的方式明確併入本文中。ISO/IEC 7816的卡尤其適用於其中必須以安全方式(其使資料極難以或不可能以未經授權之方式被讀取)來儲存資料的應用程式。小型ISO/IEC 7816卡通常用於蜂巢式電話以及其他應用程式中。如上文所提及的，當將此等記憶卡用於新應用程式中時，其可能需要在協定之現有版本中沒有的功能或命令。可參看圖1說明此情況。如圖1之上部分所示，目標為在主機側與卡側上的特定應用程式(例如在安全資料傳送或電子商務應用程式中)之間交換命令及資料。為了實施此目標，將需要在主機與卡之間傳輸該等命令，其中圖1之下部分展示了主機側上之某些軟體層及卡側上的某些軟體層。

來自主機之應用程式層之指令將被傳遞通過作業系統、檔案層及裝置(卡)驅動程式，終止於可藉以將其傳輸至卡的協定中。在卡內，指令將接著由裝置層韌體(其處理標準卡操作)加以處理並被傳遞至韌體之應用程式層。視正使用之配置而定，傳輸協定中之指令將自主機直接地交換至卡或經由所配接之讀取器或硬體交換至卡，該讀取器或硬體可具有其自身之用於自一協定轉譯至另一協定的軟體/韌體層。其中可出現問題之處為在此等層之間的轉譯中之某階段，應用程式之預期指令可能缺少在沿該路徑之某點處的相應命令。

作為一實例，卡通常係藉由使用接受來自主機系統之命令的(例如，用於USB的)硬體配接器而與PC主機相連接。然而，在主機與硬體配接器藉以通信的協定中無法使用許多特定媒體命令，即使PC主機之主機應用程式希望將此等命令傳輸至卡應用程式。

圖2展示第一主機及一卡之系統，其中卡101可直接地(例如，藉由插入槽153中)或經由某種配接器而連接至主機151。卡韌體係由FW 105加以指示。(參看圖3中之卡韌體105及讀取器韌體335將瞭解，較一般而言可將此等功能實施於硬體、軟體或此等硬體及軟體之某組合中)。此主機151之實例可為數位相機或電話。若干類型之此等卡當前正在使用中且正在開發中。卡及主機可經由若干特定協定而通信，許多該等特定協定專用於特定媒體且其可包括各種特定媒體命令。由於許多該等命令可為媒體專用的，所以可

出現以下狀況：當將在主機與媒體之間交換此命令時，於沿線之某處該命令將需要被轉譯成可能不支援特定媒體命令的另一協定。若此另一協定不具有相應命令，則主機將不能夠成功地發出該命令。

舉例而言，除其用於主機之外，其亦通常供使用者存取個人電腦上之卡。舉例而言，其通常用於一已用於(例如)數位相機之卡且希望存取儲存在個人電腦上之卡中的相片。由圖3之盒狀圖展示此情況。卡101通常經由一具有一用於卡101之插座333的卡讀取器331而與個人電腦PC 351通信，儘管在其他狀況下該卡可直接附著至PC 351。讀取器331與PC 351通常將經由協定而通信，該協定至少在某種程度上不同於由卡101用以與主機151通信的協定。讀取器331基於韌體335(或硬體、軟體或此等硬體及軟體之某組合)而將來自PC 351之命令轉譯為適用於卡101的形式，其中通常應瞭解，可視實施而由軟體與硬體之任何組合來執行此功能。圖4中示意性地展示此過程。

為給出具體實例，在圖4中，將讀取器331視為使用SCSI命令集與PC 351通信之USB裝置且將卡101視為SD卡。因為PC 351及讀取器331使用SCSI協定，所以當主機希望經由讀取器而向卡發出命令時，其發出SCSI命令集中之命令401。為將命令401傳送至讀取器，將命令401置放於USB包裝403中、沿USB連接而傳輸至讀取器，且在該處移除USB包裝。卡讀取器331將接著將來自SCSI命令集之命令401轉譯成SD集中之相應命令或多個命令405，從而允許將其傳遞至卡

101上；然而，為了使讀取器在SCSI命令集與SD命令集之間轉譯給定命令，需要在兩個集中之均等命令。因此，若主機希望藉由發出SD協定中之安全讀取命令而執行(例如)對SD卡中之安全區域的讀取，則因為不存在此SCSI均等命令，因此讀取器因為未在SCSI命令集中找到其而無法對此進行轉譯且讀取請求將被視為其中使用者無足夠存取權限之區域。對於各種卡類型之其他特定媒體命令而言，將出現相同情況，其中在讀取器所理解之協定中無發送至該讀取器的用於主機之均等SCSI命令或多個均等SCSI命令。

因此，在不改變配接器或讀取器之韌體的情況下，不能夠將此媒體卡特定命令自主機傳遞至卡，配接器或讀取器之韌體控制(host)媒體卡引入其用於與主機通信之協定的相應命令。無論何時沿通信路徑之某處需要協定變化時或當主機之作業系統不瞭解卡特徵時，即使在無獨立卡讀取器的情況下，亦可出現類似情況。

在先前技術中，此問題之方法將為改變配接器韌體以支援用於經由配接器而傳送此等指令的特定命令或在讀取器-主機協定中建立新命令，其每一者皆經定製以適於特定媒體命令，諸如對應於(例如)SD卡之協定中之發送詢問命令、接收回應命令等的彼等特定媒體命令。此等方法由於許多原因而傾向於不切實際。舉例而言，該等協定通常係基於一標準(其為了成為有用之標準而需要被廣泛接受且通常傾向於偏好較不複雜之命令集；因此，當引入新媒體且現有媒體發展時難以將各種新的特定媒體命令引入命令

集中。若替代地，改變配接器之韌體以允許特定命令傳遞韌體自身並不支援的彼等特定媒體命令，則此等配接器之每一廠商的軟體將需要接受並進行適當韌體變化。因此，引入個人電腦可利用各種特定媒體命令而不必改變讀取器之韌體或主機-讀取器協定所使用的方法將具有極大益處。

更一般言之，即使當卡直接與主機連接時，因為引入了新特定應用程式命令，所以協定將需要被更新。即使可對現有協定進行擴展及標準化以併入有新功能，此仍引入了與舊版本或彼等其他廠商之問題的相容性，從而極大地破壞了標準化協定之效用。另外，儘管該等協定可經調適以適於新應用程式，但因為出現日益增加之應用程式來擴展卡之使用，所以此可能僅為臨時解決方案。因此，類似地需要適應其中主機之作業系統不支援特定媒體命令之狀況。

【發明內容】

因此，簡言之且一般而言，本發明提供裝置(諸如，記憶卡或其他積體電路卡)可與主機交換應用程式或特定媒體命令所使用的方法及技術。主機與卡可直接地或經由讀取器或硬體配接器而通信，該讀取器或硬體配接器經由一不具有特定媒體命令之均等命令的協定而傳輸特定應用程式命令。舉例而言，在一實施例中，即使卡讀取器不支援此命令，PC亦可經由卡讀取器而讀取SD(安全數位)記憶卡之安全資料區域。

更具體言之，主機形成具有一在沿主機與卡之間的通信

路徑而被支援的協定中之命令的指令，同時實際媒體或特定應用程式命令內嵌於該指令中。在例示性實施例中，特定媒體命令內嵌於指令之資料部分中。接著卡接收協定(該卡經由該協定而與主機通信)中之指令並將其轉譯為應用程式之協定。內嵌命令以透明方式傳遞；例如，當傳輸協定實際上可含有特定媒體命令時，該傳輸協定正好傳遞至卡上其認為是資料之處。一旦在卡中，其便識別出實際命令內嵌於指令中且繼續提取該指令。此係藉由判定命令之資料部分含有一簽名而進行：若發現簽名，則提取並執行內嵌命令；若未發現簽名，則執行傳輸協定之命令。在例示性實施例中，簽名及內嵌命令置放於資料部分之第一區段中。在裝置側實施轉換，其中提取實際命令，且在主機側中，於裝置驅動程式層級或檔案系統層級內嵌命令。裝置驅動程式層級實施將命令定址至特定邏輯塊位址，同時在檔案層級處可使用任何邏輯塊位址。儘管裝置驅動程式層級實施需要較少額外耗用，但在許多作業系統中其可能需要使用者所缺乏的存取權限。

本發明之各種態樣不僅限於卡協定具有在主機與硬體卡讀取器之間所使用的協定中無均等命令的命令之狀況，而且限於其中中間協定需要載運其所缺乏之命令的其他狀況。第一組實施例考慮其中卡可以其自身協定直接與主機通信的狀況，但卡(或其他裝置)具有一具有需要使用一額外命令集之額外特徵集的應用程式。此等命令並非由主機與卡進行通信所使用的標準協定之部分，或並不由主機上之

標準作業系統加以支援。因此，需要特定構件以在卡上之應用程式(可於韌體層級實施)與主機上之應用程式(可於軟體層級實施)之間來回地傳遞此等命令。在本發明之原理態樣中，應用程式命令內嵌於中間協定中之指令的資料部分中，在此狀況下該中間協定為卡協定。在本發明之另一態樣中，此係藉由具有與開啟、關閉及管理自主機應用程式至裝置之路徑相關的命令集來達成。

第二例示性實施例係實施於裝置(卡)驅動程式層級且使用一指令，該指令指定特定邏輯塊位址(LBA)、卡傳遞(CPT)模式LBA，該卡傳遞(CPT)模式LBA具有一特定簽名以通知該卡一特定命令內嵌於區段中。此可被實施為卡韌體之變化，以使得其支援CPT模式。此消除了根據每一特定媒體命令而改變配接器之韌體的需要，使得卡可在任何(例如)USB讀取器或配接器處執行而不必進行特定媒體調適(或改變主機之作業系統(OS)，使得卡可在任何OS下使用)。卡韌體繼續執行(honor)正常讀取命令/寫入命令。在使用至特定LBA偏移之讀取命令/寫入命令的情況下，檢查一簽名以判定資料是否含有內嵌媒體卡特定命令。協定可實施於任何媒體卡之韌體中。因此，在不改變讀取器韌體的情況下，不需要配接器或USB或其他讀取器或主機OS的韌體改變。

第三組例示性實施例實施於主機側之檔案層級。特定媒體命令再次內嵌於中間協定中，但不是依賴於硬體詳細說明及參考特定協定，而是將內嵌之命令及任何隨附資料全

部置放於檔案之資料部分中，接著將該檔案傳送至媒體，其中韌體再次提取實際命令。在此等實施例中，主機僅告知檔案系統將檔案寫入至記憶體裝置，其中裝置特定命令再次內嵌於資料部分中。在使用至任何LBA之讀取命令/寫入命令的情況下，與特定邏輯位址相反，裝置檢查一簽名以判定資料是否含有內嵌媒體卡特定命令。例示性實施例將此簽名置放於與寫入命令相關聯之任何檔案中的第一資料區段中。若發現適當簽名，則提取內嵌命令；若未發現簽名，則將該命令解釋為中間協定之標準命令。以此方式，本發明之檔案層級實施例允許繞過在裝置層級應用程式中使用特定邏輯位址而出現的權限問題，因為在作業系統中通常允許將檔案寫入至裝置。

在所有例示性實施例中，將內嵌命令置放於中間協定中之寫入命令的資料部分中。舉例而言，裝置或特定應用程式協定中之(例如) N 個區段的寫入命令將由具有 $(N+1)$ 個資料區段之中間協定中的寫入命令組成，第一區段含有實際的內嵌寫入命令且剩餘 N 個區段含有待寫入之實際資料。更一般而言，可將命令(及簽名)連同待由內嵌命令傳送至裝置的任何資料一起置放於接受相關聯資料部分之任何命令中。為實施裝置或特定應用程式協定中需要將資料自裝置傳送至主機的讀取命令或其他命令，例示性實施例使用一對命令：第一命令將再次為具有一資料區段之中間協定中的寫入命令，其對應於內嵌讀取命令(不具有與內嵌讀取相關聯的實數資料)；自裝置至主機之實際資料傳送接著由

第二命令(經定址至與第一命令相同的邏輯位址之主機讀取)實現。

本發明之額外細節、特徵及優點自以下描述將變得顯而易見，以下描述應結合隨附圖式來進行。

【實施方式】

在一主要態樣中，本發明允許在主機上之應用程式(通常為使用卡上之應用程式的軟體)與卡上之應用程式(通常為將此等額外特徵實施為卡上之特定應用程式的卡韌體)之間來回地傳遞並非標準卡協定之部分(或不由主機上之標準OS而支援)的命令。例示性實施例藉由將特定命令內嵌於任何主機所支援之標準命令(諸如寫入)中而實現此操作。此允許諸如[先前技術]段中所描述之記憶卡在所有標準主機裝置及PC上被識別、建置介面及使用，同時允許未經支援之額外非標準特徵作為一標準而併入於系統中。

作為具體實例，參看關於圖2至圖4於先前技術中所述之特定狀況，此等方法允許專用於SD卡之命令(諸如安全讀取)經由缺乏均等命令之USB協定而傳輸。一般而言，此係藉由形成一由存在於傳輸協定中之命令部分及一其中內嵌有特定媒體命令的資料部分組成之指令而實現。舉例而言，因為寫入命令對於記憶卡系統而言為基本命令，所以其將存在於傳輸協定中且具有相關聯之資料部分。將實際特定應用程式命令與簽名一起置放於資料部分中以指示將執行特定媒體命令。亦將與待傳送至裝置之實際內嵌特定媒體命令相關聯的任何資料置放於傳輸協定之命令的資料

部分中。對於SD/USB實例而言，為在USB協定中將十個區段之寫入讀取發送至SD卡，在包含一命令及十一個資料區段的USB協定中發出一指令：第一資料區段含有實際SD安全寫入命令以及一簽名及相關資訊，且隨後為待寫入之資料之十個區段。當在卡中被接收時，韌體提取實際安全寫入命令、檢查簽名並執行資料之讀取。

對於讀取(及將資料自裝置傳送至主機之其他命令)而言，中間協定中之寫入命令可再次用於載運內嵌讀取命令，但因為此不提供待傳送回至主機之資料，所以現其與中間協定中之第二讀取命令成對。更具體言之，該對中之第一命令將為來自主機之具有至邏輯塊位址LBA=XYZ之內嵌讀取命令的寫入(在中間協定中)。在裝置驅動程式層級實施中，邏輯塊位址LBA=XYZ將為特定位址，同時對於檔案層級實施，此可為任何位址。接著該對中之第二命令為至此同一位址LBA=XYZ的標準主機讀取。接著將特定資料傳送至主機。

更一般而言，本發明允許藉由將所要命令與任何隨附資料一起內嵌於可由系統使用之中間或傳輸協定之指令中而在主機與卡之間交換應用程式或特定媒體指令。在執行於一側的主機上之特定應用程式與另一側的特定卡上之相應應用程式之間剖析(resolve)特定命令。在沿傳輸路徑之中間點處，指令呈現為具有中間協定之正常命令，以透明方式傳遞之實際特定應用程式命令。在例示性實施例中，此係藉由使用接受一資料部分之傳輸協定中的命令(諸如寫入

命令)及將實際應用程式或特定媒體命令內嵌於此資料部分中而進行。(在某些方面，此為在2002年9月26日申請的同在申請中之共同讓渡之美國專利申請案第10/256,689號中所述的本發明之擴展，該案以引用的方式併入本文中。)當指令(傳輸協定中)到達卡時，檢查資料部分以察看其是否含有適當簽名，且若含有適當簽名，則提取內嵌命令。圖5中對此進行了示意性展示。

圖5為根據本發明之一特定實施例如何將一特定應用程式命令內嵌於協定中之示意性表示，該協定用於將協定自主機應用程式傳遞至應用程式之卡側。考慮其中應用程式或特定媒體指令具有一命令部分、CMD及某相應資料之狀況。舉例而言，CMD可為特定應用程式寫入。為傳輸此指令，主機側將採用傳輸協定中之命令，稱其為CMD'，例如寫入命令，其具有一相應資料部分。任何置放於此指令之資料部分中的命令將簡單地傳遞；因此，若將實際之預期特定應用程式命令CMD(連同其相應資料一起)內嵌於此資料部分中，則即使在傳輸協定中未識別出命令CMD其亦將被傳輸至卡。一旦在卡中，命令CMD將需要自資料部分中提取。此亦係藉由使卡傳遞簽名(CPT簽名)而置放於資料部分中而進行：當傳入之指令到達卡上時，檢查資料部分中之簽名，且若發現簽名，則接著提取實際命令。

在圖5中，於510處展示含有內嵌命令之傳輸協定中之指令。命令部分511具有命令CMD'且可將其視為實際上將不被執行但用來將資料部分513傳送至卡側的一種"虛設"命

令。可將資料部分513分成具有簽名及實際命令CMD之第一部分513a及對應於CMD之任何資料的第二部分513b。舉例而言，若CMD為一類型之寫入命令，則513b將為待寫入之資料，而若其為(例如)狀態檢查，則將無513b。(如下文進一步描述的，較可能為內嵌讀取命令，因為(在此實施例中)其涉及發送傳輸協定之區段511中的寫入命令CMD'以及資料部分之區段513a中的內嵌讀取命令CMD及非實數資料部分513b。)在卡側，接著檢查513a之簽名且若未發現簽名，則將命令CMD'與513之全部作為資料加以執行。若發現簽名，則提取具有包含命令CMD之命令部分531及具有資料部分533中之任何相應資料的實際指令530。(在命令部分511中具有足夠空間之諸如SCSI協定的協定中，亦可將CPT簽名、命令CMD或其二者內嵌於傳輸協定之命令部分而非資料部分中。)

返回參看圖1及本發明之更抽象描述，本發明允許應用程式之主機側與卡側通信，即使其需要在沿路徑之某處使用一不適用於該應用程式之協定來彼此通信。此可為當配接器不瞭解卡特徵(如先前技術中所描述)而且主機作業系統不瞭解卡特徵時的狀況。此解決了具有或不具有位於主機與卡之間的主機配接器之系統的媒體或特定應用程式命令之發出。另外，其允許其中卡協定可不僅在卡形式因數之間不同而且在相同形式因數之各種卡之間不同的狀況，因為由於此等卡正變得過載有卡與卡之間可不同的額外功能。本配置允許其中卡配接器及主機OS與特定卡協定無關

的系統，其中在執行於一側之主機上的特定應用程式與另一側之特定卡上的特定應用程式之間剖析特定命令。

在主機側，本發明可實施於檔案系統層級(其中在檔案模式中應用程式正使用主機檔案系統並寫入至檔案)或可實施於裝置驅動程式層級(在一使用裝置驅動程式之模式中)。下文給出兩種版本之特定實例。哪一類型之實施例係較佳的通常將取決於諸如特定應用程式之詳細說明及作業系統之細節。舉例而言，在裝置驅動程式層級之實施通常較簡單且需要較少軟體/韌體額外耗用，但使用至邏輯位址之直接寫入；然而，此直接寫入需要使用者將缺乏之管理員權限。藉由替代地，使用檔案系統層級之實施可避免此等複雜性，但以額外額外耗用為代價。

將藉由三個特定實施例之更詳細描述來說明本發明。所有此等實施例允許在不改變用以支援新特徵之主機卡協定或同時藉由不支援額外協定之中間軟體/硬體而使用其的情況下，待實施於卡中之特定非標準特徵用於主機上。第二及第三組實施例描述實施於主機側(分別在裝置驅動程式層級及檔案系統層級)之兩個方法，同時第一實施例描述卡側實施。此第一實施例補充第二及第三實施例並描述卡側如何操作且亦將用於提供對所有實施例之更一般的細節。如下文中更多描述的，此係使用標準可用命令以包封(envelop)非標準命令而進行。對於此等狀況中每一者而言，例示性實施例選擇處處皆被支援之標準讀取寫入。應注意，在此等特定實施例中，卡無需識別檔案之概念，因

為在所有此等實施例中卡瞭解一至特定邏輯塊位址(LBA)之寫入。此LBA可為此目的而被具體地配置(或再配置)(如在裝置驅動程式模式中)，或可為任何位址(在檔案驅動程式模式中)。主要不同之處在於：在主機側(其中在檔案模式中)，應用程式正使用主機檔案系統並寫入至檔案，而在第一模式中，其正使用裝置驅動程式且直接寫入至LBA。

在下文中，於若干特定實施例之實例中提出本發明之各種態樣。舉例而言，該等實施例已將特定媒體命令內嵌於寫入命令之資料部分中；更一般而言，可將媒體或特定應用程式命令內嵌於其他命令中。同樣，[先前技術]中之論述係基於具有特定協定域之特定硬體配置；亦即，其中PC/主機使用一在主機與讀取器之間的協定及在讀取器與卡之間的第二協定而經由硬體讀取器與卡進行通信。此等實施例皆可以若干方式來一般化及擴展。

[先前技術]之論述集中於其中卡與PC經由讀取器而通信之實例。更一般而言，當(在主機(例如，PC)與最終目標之間的路徑中之某處)使用不具有需要傳輸之一或多個命令之均等命令的協定時，本發明之各種態樣適用。如所提及的，除當硬體卡讀取器以一在卡協定中不具有主機希望發送至卡之特定媒體命令之均等命令(例如，SD卡中之安全讀取)的協定(例如，USB/SCSI)與主機通信時出現的問題之外，當卡接著直接地附著至主機時亦可出現問題。在更複雜之狀況下，可存在若干中間協定，在該狀況下可使用若干內嵌層，如在第二實例中將看到的。

作為當不需要存在獨立卡讀取器時的特定實例，存在具有兩組接點之卡，一組供USB埠使用且另一組供一標準組(MMC、SD、CF等)之卡接點使用。(在皆於2004年4月16日申請之美國專利申請案第29/203,693、US 10/826,801及US 10/826,796號中描述此等雙接點卡之實例，該等案皆以引用的方式併入本文中。)在使用"正常"卡接點的情況下，卡可以允許特定媒體命令之卡特定協定與第一主機通信。在使用第二組接點(此處，USB)的情況下，卡可以缺乏特定媒體命令之均等命令的另一不同協定與諸如PC之另一主機通信。在此狀況下，第二主機以其中根據本發明而內嵌特定媒體命令的其他協定與卡通信：例如，使用USB接點，卡可在USB埠處直接連接至PC，藉此在使用內嵌於如上所述SCSI協定中之任何SD特定命令的情況下，其以SCSI協定進行通信。

如所提及的，在尤其有關的一組實施例中，主機與卡以卡協定進行通信，但主機應用程式可希望與卡上之應用程式交換命令及資料，其中在該卡之一層中不瞭解卡的"通常"記憶體功能。舉例而言，SD記憶卡可為自SD特定韌體層隱藏的裝置應用程式層分割。主機與卡將使用標準SD協定進行通信，但隱藏之裝置應用程式可藉由將特定應用程式命令內嵌於SD協定內而與相應主機應用程式進行通信。第一例示性實施例為此類型。

例示性實施例將特定媒體(或特定應用程式)命令內嵌於寫入命令內。寫入命令係用作將命令內嵌於中間協定之資

料部分內的實例且寫入命令固有地具有一資料部分。更一般而言，不僅可類似地利用寫入命令而且可類似地利用接受一資料部分之任何命令。由於寫入為協定中用以傳送資料之基本指令，所以下文之例示性實施例將繼續根據寫入進行描述。此外，儘管為簡單起見在所有實例中媒體或特定應用程式命令內嵌於資料部分之開始處，但更一般而言，其可置放於資料部分中之其他處且由簽名來識別。(如上文所提及的，更一般而言，簽名、應用程式命令或其二者亦可內嵌於協定(在其命令部分中具有足夠空間)中之傳輸協定(諸如 SCSI 協定)的命令部分中。)

本發明涉及之問題可以更抽象之層次而呈現。考慮其中主機應用程式需要以一具有命令集(α 、 β 、 γ 、...)之協定與裝置應用程式進行通信，但在通信路徑中之某處需要使用具有命令集(A、B、...)之另一協定。在[先前技術]之實例中，(α 、 β 、 γ 、...)將為SD協定且中間命令集(A、B、...)將為SCSI協定。在下文之論述中，(α 、 β 、 γ 、...)將為用於特定應用程式(諸如隱藏分割)之命令集且(A、B、...)將替代地為卡命令集。只要存在對應關係 $A \leftrightarrow \alpha$ 、 $B \leftrightarrow \beta$ 等，便不存在問題。當存在不具有中間協定中之均等命令： $? \leftrightarrow \gamma$ 的媒體或特定應用程式命令 γ (例如，SD協定中之安全讀取)時，出現困難。在先前技術下，將需要對於每一此 γ 引入均等於 γ 之新命令(例如， $C \leftrightarrow \gamma$)。實情為，根據本發明，將特定媒體命令內嵌於中間協定中之現有命令中($A(\gamma) \leftrightarrow \gamma$ ，其中 $A(\gamma)$ 表示一內嵌有 γ 之寫入(例如)命令)，而非引入新命令。

一般而言，將一命令(稱其為F)引入(α 、 β 、 γ 、...)協定中以開啟供應用程式之主機側與裝置側通信的管線。一旦該管線開啟，便可接著發送媒體或特定應用程式命令。此可表示為 $A[F(\gamma)] \leftrightarrow \gamma$ 。將在第一實例中發展此技術。此允許處理命令失敗及其他細微之處。具有命令失敗之問題在於：由於據中間協定所知，其正發送一簡單寫入命令，故中間協定不瞭解內嵌命令之性質。若內嵌命令為(例如)讀取，則PC將不會知道在讀取失敗的情況下已發生什麼。在中間協定中，將其視為寫入失敗且不將其解釋為內嵌讀取之失敗；因此，主機將不會理解待執行之適當復原步驟。藉由開啟在應用程式之主機側與卡側之間的管線，可更易於追蹤並處理此等困難。

第一例示性實施例

第一例示性實施例係實施於主機側的檔案層級且將用於說明本發明之各種態樣；在卡側，不管主機側是實施於檔案層級還是實施於裝置驅動程式層級，實施很大程度上為相同的。當參考此實施例之特定應用程式時，其將在記憶卡上之隱藏分割之內容中被給出，其中記憶體分成可公共存取部分及一私用(或"隱藏")部分。在於2004年12月21日申請之美國臨時專利申請案60/638,804(該案以引用的方式併入本文中)中描述此配置，儘管多數細節並未被如此限制。首先將描述主機側部分(詳言之，裝置驅動程式)，隨後描述卡細節。在以下專利申請案中描述隱藏分割之使用的其他實例，其全部以引用的方式併入本文中：在2005年2月25日

申請之11/067,298；在2003年11月10日申請的10/703,471；在2004年7月26日申請之10/899,260；及在2005年2月2日申請的11/050,013。

圖6為展示裝置側之例示性結構的示意性方塊圖。在卡1001側，裝置應用程式或多個應用程式1023各自提供用於主機側1051之主機應用程式或多個應用程式1053至對於此特定形式因數的非標準卡特徵(此處為隱藏分割)的閘道或"管線"。下文在圖7中更多地描述主機側層的情況。在卡側，裝置應用程式1023供應分割存取(讀取及寫入)、狀態及其他裝置應用程式資訊。裝置應用程式1023可經由下文所述之例示性協定而與主機應用程式1053通信且經由媒體層1025而使用卡之系統及記憶體資源。在記憶體1011處指示記憶體自身且其通常將為NOR或NAND種類之快閃記憶體，儘管其亦可為其他非揮發性記憶體，包括2004年5月7日之美國專利申請案10/841,379(該案以引用的方式併入本文中)中所述之彼等記憶體。FE(前端)層1021為韌體層，其處理來自主機側之命令(在諸如CF、SD、MMC等協定之卡類型介面中)。BE("後端")層1027為韌體層，其處理對記憶體1011之管理，包括讀取及寫入至記憶體。媒體層1025將裝置應用程式韌體1023與BE層1027相連接且亦對應用程式韌體與控制器之RAM資源建置介面，該控制器未在此圖中具體展示。

當自主機1051發送指令時，最初在裝置1001上於前端(FE)層1021處接收到該指令。該命令將在於主機與裝置之

間所使用的中間協定(例如，SD協定)中。當命令到達FE層1021時，檢查命令傳遞簽名：若未發現簽名，則將該指令視為來自中間協定之標準指令且該指令可存取記憶體或以標準方式來執行該指令；若發現簽名，則提取內嵌特定應用程式命令且將其傳遞至適當裝置應用程式1023上以供執行。當實施於裝置驅動程式層級(主機側)(如在下文之第二例示性實施例中)時，將命令定址至特定邏輯位址，同時當實施於檔案層級(如在此實例與以下第三實例中)時，可將命令定址至任何邏輯位址；如自裝置側所瞭解的，兩個實施在很大程度上為相同的。

一旦發現簽名且提取內嵌命令，便使隱藏分割(在此實例中)經由位於卡1001上之相應裝置應用程式1023而可用於主機應用程式1053。主機與裝置應用程式將使用一將覆蓋主機側與卡側之間的讀取及寫入之協定以及報告分割狀態而進行通信。在此例示性實施例中，裝置控制對應用程式之權限。當裝置確認主機應用程式之憑證(credential)時給予該等權限(讀取、寫入)，且一經核準，裝置應用程式便經由主機/裝置協定而向分割授予讀取及寫入權限。一旦已確認主機應用程式，便接著假定裝置正運作於信任環境中。

如所提及的，第一例示性實施例將實施於檔案層級，其中內嵌命令置放於寫入檔案之第一資料區段中。實施並不依賴於對任何特定邏輯位址之參考，而是依賴於檢查第一資料區段的特定圖案或簽名。分割中之檔案係由主機加以管理，其意味著在隱藏分割實例中，主機應用程式為隱藏

分割中之檔案系統的所有者。一旦主機應用程式獲得對應應用程式之權限，其便完全控制內容。

由於歸因於作業系統(OS)之存取權限問題而通常不能夠讀取隱藏分割之FAT，所以主機應用程式1053將實施檔案系統層。如同對隱藏分割之任何存取一般，僅在至應用程式之管線開啟之後FAT才為可用的。(下文將界定及闡述由主機應用程式進行的命令傳送。)由大容量儲存裝置或抽取式驅動程式層級之OS所瞭解的，主機應用程式將位於卡之非隱藏或公共分割上。若干此等主機應用程式可位於此區域中且使用者將知道執行哪一個主機應用程式、將在主機側執行哪一個主機應用程式。

參看圖7更詳細地描述主機側軟體層。裝置驅動程式1055具有兩個任務。其充當匯流排驅動程式並執行由(中間)協定及檔案系統1057支援的標準用戶端應用程式之IO操作。此外，其亦實施應用程式及裝置或特定應用程式協定。標準IO API及實施可為卡家族之全部所共有(例如，標準SD卡操作)。應用程式或特定裝置API及實施本質上為特定協定。圖7展示階層式結構之實例，該階層式結構賦能共同部分之碼共用及不同特定應用程式協定之間的簡單可移植性。

除主機應用程式層1053及檔案系統層1057之外，主機結構現亦併入有專用於裝置應用程式之層1059。對於"標準"裝置操作(未併入有特定應用程式操作)而言，不使用層1059且層1053、1057及1055將大體上如先前技術中一般而操作。當將使用特定應用程式指令時，裝置應用程式層1059

將被插入於檔案系統1057與主機層1053之間，以格式化及內嵌特定應用程式命令。

裝置驅動程式1055可包括介面及管線子層以提供常用功能性且可配合所有特定應用程式協定之最小變化一起使用。主機之應用程式應根據特定應用程式而與協定層匹配。介面子層將呈現一組標準裝置功能。舉例而言，此等功能將包括：硬體初始化及組態；驅動程式開啟及關閉例行程序；讀取及寫入操作；及擦除。其亦將包括用於進行適當通信方法並因此而初始化管線之功能。

當傳輸諸如圖5之510的指令時，在某些狀況下，作業系統(OS)可將其分成若干片，可能甚至插入其他指令的片。在此等狀況下，檔案系統1057及裝置驅動程式1055可分解指令510，使得命令部分511處於不附著至資料部分513直至附著至所有資料部分513中的任一情況下，其中剩餘資料部分接著在一或多個隨後傳送中被傳送。在此等狀況下，對剩餘資料部分之隨後傳送將沒有區段513a之簽名部分並表現為標準命令。對於在讀取操作中自卡傳送至主機之資料而言，亦可發生此分割。在其中已知OS將不分裂指令或OS可受到足夠控制以使得其可經指示以不進行此操作的狀況下，可簡化程序。首先參看圖10A之流程及圖12之相應狀態機圖對此進行論述。隨後將參看圖10B論述更一般之狀況。此外，為論述之簡單起見，參看圖10A之論述大體上將基於僅有單一應用程式之狀況，儘管通常將有若干不同的此等應用程式在作用中。圖10B之更一般狀況將使此更明確。

對於第一例示性實施例，將簡單隱藏分割協定用於說明。將記憶體1011分成兩個分割。第一分割為產品標準分割。標準主機僅瞭解此分割。第二分割為僅可經由特定應用程式命令而存取之隱藏分割。標準主機不瞭解此分割且沒有辦法存取該分割。該實例之隱藏分割協定界定了裝置與應用程式進行通信而使用的五個功能：

- 1.開啟隱藏分割：用於鑑認使用者並賦能讀取及寫入操作。

- 2.關閉隱藏分割：去能讀取及寫入操作。

- 3.獲得隱藏分割狀態：傳回圖8中所示之資料。

- 4.隱藏分割讀取：若分割開啟，則自隱藏分割進行讀取。若分割關閉，則傳回一錯誤。

- 5.隱藏分割寫入：若分割開啟，則寫入至隱藏分割。若分割關閉，則傳回一錯誤。

協定層提供實施安全協定所需之額外API。在隱藏分割之特定實例中，除讀取及寫入之外，協定還需要三個服務：開啟分割、關閉分割、獲得分割狀態。下文在表1中展示此等命令，其中結合圖8對其進行進一步論述。

管線子層負責根據媒體命令傳遞原理而與卡進行的通信。如先前所描述，媒體命令傳遞協定中之主要思想為將特定命令(並非標準產品規格之一部分的命令)內嵌於標準讀取及寫入操作內。將該等命令內嵌於標準讀取及寫入操作中使得能夠與標準主機及驅動程式一起工作。

在例示性實施例中，所有命令將藉由將一寫入命令發送

至媒體卡之某一LBA而初始化。特定應用程式寫入命令將包括多個區段傳送，其中第一區段保存命令之自變數且剩餘資料塊保持相關資料(若存在)。將在兩個部分中執行讀取命令：

1. 藉由首先將一寫入命令發送至一具有描述讀取命令之性質的適當資料之選定LBA而初始化該讀取命令。
2. 在寫入命令將卡應用程式設定於正確傳送狀態之後，自寫入命令將讀取命令發送至選定LBA以執行自卡至主機的實際資料傳送。

所有命令首先發送一具有圖9所示區段格式的寫入命令。當主機應用程式決定發送寫入命令時，所傳送之第一區段將具有上述格式。自主機發送之下一資料塊將具有卡之實際資料。若主機應用程式決定初始化讀取命令，則其首先將一寫入命令發送至具有上述單一區段格式之選定LBA(LBA=LBA_XYZ)並接著將一讀取發送至同一LBA。在下文進一步論述之圖10A及圖10B之流程圖中對此進行展示。

若干可能之實施可用於媒體命令傳遞協定。此段之實例使用檔案系統，例如Windows檔案系統操作。下一段之實例在裝置驅動程式層級直接與裝置通信且不具有檔案系統額外耗用。另一實施可用於袖珍型PC(PPC)。在此最末狀況下，讀取及寫入操作可直接存取PPC儲存驅動程式。由於此處不存在用於袖珍型PC裝置之標準儲存裝置驅動程式，所以不保證此方法將適用於所有PPC裝置。用戶端應用程式廠

商應使用其意欲與之一起工作的PPC裝置組對此方法進行測試。

返回本實例之檔案系統實施，對於其中OS將在單一傳送中傳送完整指令的狀況而言，管線子層藉由自(如圖10A中之LBA_XYZ所指示的)相同檔案位置進行讀取及寫入至該相同檔案位置而初始化對命令傳遞LBA的讀取及寫入操作。此無需如下一段之實例中一般為預定位址，而是可為任何邏輯塊位址。對於此實例而言，實施使用Windows之標準檔案系統API-CreateFile、ReadFile、WriteFile及SetFilePointer。可將實施集合分成四個或五個部分：預先之要求；通信管線之建置；檔案長度之驗證；應用程式寫入命令；及若將讀取資料或將其他資訊傳回主機，則為讀取命令。

在檔案層級實施中，檔案系統用於不知不覺地傳輸內嵌之特定應用程式命令。由於實際內嵌命令可不同於檔案系統認為其正發送的在中間協定中之命令之性質，所以若不採取適當步驟則可出現衝突。舉例而言，若出現錯誤，則因為當實際錯誤可與內嵌命令(例如，讀取命令)在一起時檔案系統會將此視為中間協定中之載運命令(寫入命令)中的錯誤，故錯誤復原可為複雜的。類似地，若檔案系統採用快取，則可導致不一致。

另一問題可由於檔案分段而產生：因為例示性實施依賴於自相同檔案位置(圖10A之LBA_XYZ)進行讀取及寫入至該相同檔案位置，所以應將整個檔案儲存至記憶體中一連

續群組之區段。若待在LBA_XYZ處開始儲存之資料大於可用片段(segment)，則檔案系統將會將其分開。因此，特定應用程式命令之最大資料長度應不大於最小OS檔案配置單元(叢集)，使得所使用之邏輯位址(LBA_XYZ)與一具有足夠空間之邏輯塊位址相關聯。

在Windows實例中，檔案系統實施之預先之要求需要一含有至少128個連續區段的標準檔案。為確保此檔案之存在，將其建立於格式化之磁碟機上。因此，若主機應用程式廠商選擇此方法，則其應佈署具有該應用程式之通信檔案。在此實例中，設計假定名為"FilePipe"之檔案(具有下文指定之某些屬性)存在於裝置使用者區域中。若此檔案不存在，則驅動程式將試圖建立該檔案。若其失敗，則應用程式可要求使用者格式化磁碟機並在格式化磁碟機上重新安裝應用程式。

為建置通信管線，裝置驅動程式1055較佳開啟具有以下屬性之通信檔案：

未經共用

未緩衝

經隱藏

未經共用之狀態確保僅主機應用程式可存取檔案，以防止其他應用程式存取命令傳遞LBA。未緩衝狀態指示系統開啟無中間緩衝或快取之檔案。因此，至/自此檔案之任何寫入或讀取導致不同於快取版本的至/自卡之讀取命令/寫入命令。(在某些狀況下，為確保將刷新至卡的檔案，可能

有必要設定"未緩衝"旗標與"寫傳遞"旗標。)由於實際應用程式或特定媒體命令內嵌於另一協定之命令中，所以當存在(例如)寫入錯誤時，主機將此解釋為與中間協定中之寫入命令有關的錯誤而非與內嵌命令有關之錯誤。未緩衝狀態保持資料傳送與預期的內嵌命令相關。隱藏狀態暗示最終使用者將此視為系統檔案。

下一實施步驟為驗證檔案含有至少128個連續區段。

對於應用程式之寫入命令(隱藏分割實例中之安全寫入)而言，沿上文所描述之線預備寫入緩衝器。接著對檔案指標重設以確保讀取及寫入操作係對於相同LBA(=LBA_XYZ)而言。接著藉由調用具"FilePipe"名稱之"WriteFile"命令而將緩衝器發送至裝置。

對於應用程式之讀取命令(隱藏分割實例中之安全讀取)而言，首先重設檔案指標，接著執行"WriteFile"命令以發送命令緩衝器。接著再次重設檔案指標以確保對相同LBA執行寫入及讀取操作。接著執行"ReadFile"以自裝置獲得安全資料。

返回至卡側，裝置應用程式1023將管理隱藏分割(在此實例中)，從而允許擱置於主機應用程式確認時的讀取及寫入。對於隱藏分割階段，資源及記憶體存取係經由媒體層1025而進行。媒體層1025將所有裝置應用程式資料傳送導引至記憶體1011上之隱藏分割。

如上所述，裝置應用程式1023經由規則卡協定讀取命令/寫入命令而自主機應用程式1053接收其特定命令。舉例而

言，在將SD卡協定視為例示性實施例的情況下，經由SD寫入命令而在命令區段中傳遞每一命令索引及自變數。當期待來自裝置應用程式之讀取命令時，接著主機應用程式首先將發送寫入命令(經由SD寫入命令)，接著發送實際讀取命令(SD讀取命令)。在傳遞命令區段之寫入命令期間，將讀取命令內容儲存於(例如)專用於裝置應用程式之RAM空間中。當接著為讀取命令時，裝置應用程式將接著載入所儲存之讀取內容並傳送所請求之資料。所儲存之讀取內容將繼續保持原狀直至其被另一讀取內容替代為止。此將使得主機能夠執行讀取重試而不發送另一寫入命令+命令區段。

因為多數協定命令充當讀取命令/寫入命令，所以可藉由讀取及寫入至某些LBA而傳送作為特定應用程式命令之新命令。在例示性實施例中，所有命令係藉由將一寫入命令發送至媒體卡之某一LBA而初始化。特定應用程式寫入命令將包括多個區段傳送，其中第一區段保持命令之自變數且剩餘資料塊保持相關資料(若存在)。

如上所提及的，在兩個部分中執行讀取命令：

1. 藉由首先將寫入命令發送至選定LBA而起始讀取命令，其中以適當資料描述讀取命令之性質。
2. 在寫入命令將卡應用程式設定於正確傳送狀態之後，將發送自寫入命令至選定LBA的讀取命令以執行自卡至主機的實際資料傳送。

返回至圖9，所有命令必須首先發送一寫入命令，其中圖

9展示此處自變數資料區段之例示性格式。如圖9中所示，位元組0至31保持應用程式傳遞簽名(例如，"Pass Through Mode Supported")，位元組32保存應用程式之ID，位元組33保存內嵌命令之應用程式命令操作碼索引且以應用程式命令自變數資料來填充剩餘區段。

當在此實施例中主機應用程式1053決定發送寫入命令時，所傳送之第一區段將具有圖9之格式。自主機1051發送之下一資料塊將具有用於卡的內嵌命令實際資料。若主機應用程式決定初始化讀取命令，則其首先將寫入命令發送至具有上述單一區段格式之選定LBA且接著將讀取命令發送至相同LBA以實現讀取。

主機應用程式1053負責根據自裝置所見之通信流程(如圖10A中所示)而初始化所述命令。為進行此操作，主機應用程式控制卡協定之讀取及寫入操作的目標LBA，其將載運如上所述之內嵌命令。儘管僅描述單一LBA處之單一命令，但對於多個LBA而言，可發生圖10A(及圖11A至11C、圖12等)中所展示之過程，因為該等LBA可皆依賴於傳遞模式簽名。(下文參看圖10B更詳細地描述此狀況，其中不同LBA係以一下標加以指示， LBA_i)。因為裝置能夠連續處理多個檔案，所以特定LBA僅取決於裝置之狀態機(如下文參看圖12所論述的)。

當接收到至選定LBA之寫入時，該過程開始於步驟1101，此處認為 $LBA=LBA_XYZ$ 。接著檢查(1103)簽名以察看其是否為應用程式傳遞簽名：若並非應用程式傳遞簽

名，則卡進行標準IO操作(1105)；若為應用程式傳遞簽名，則進入CMD傳遞模式。步驟1109判定命令方向(如自裝置所見)是否為出(OUT)。若方向為出(例如，至裝置之記憶體之寫入)，則執行(1111)該命令。

若命令方向不為出，則在1113處，記憶體裝置等待讀取對之第二命令(應用程式讀取請求)。若替代地，寫入到達(1121)，則在返回等待狀態(1113)之前轉至標準IO操作(1123)。若接收到讀取命令，則檢查(1117)LBA且若其係至選定LBA，則其為讀取對之第二命令且執行(1119)讀取；若LBA不匹配，則替代地，轉至1123。

存在一個以上可能之實施方式且實施細節根據目標平臺而改變。一普通方式(其應用程式於基於Windows NT之桌上型電腦及上文關於主機側所描述的)為使用虛擬檔案及標準Windows檔案系統API。另一方式為使用Windows SCSI傳遞API而直接初始化卡協定之(例如，SD)讀取及寫入操作，如下一實例中所描述的。此方式亦應用程式於基於Windows NT之桌上型電腦，但其需要如上文關於卡驅動程式層級之實施而論述的管理權限。

關於卡韌體與應用程式韌體之間的命令介面，在第一步驟中，卡FW接收一特定應用程式命令。在驗證卡傳遞模式支援簽名之後，卡之FE層1021將剖析與至應用程式ID及命令OP-Code之選定LBA的寫入命令一起的自變數資料區段。特定應用程式命令解譯器將調用具有指向主機資料緩衝器之指標的應用程式介面例行程序。在第二步驟中，當

開始特定應用程式命令序列時，應用程式FW必須設定其自身狀態機以處理應用程式命令序列且亦設定一旗標以指示至選定LBA之下一讀取命令/寫入命令係用於應用程式FW。每當將讀取命令發送至卡(至選定LBA)時，卡之FW將首先檢查此旗標以區分此特定應用程式讀取命令與至此LBA的規則讀取命令。

現將描述隱藏分割應用程式之協定的實例。圖11A至11C說明與圖10A相同之某些過程，但根據主機與裝置之間的系統匯流排上之活動進行說明。圖11A中展示鑑認及安全分割FAT讀取之過程。首先，主機將使用者名稱+密碼發送至卡以供鑑認。一經鑑認，卡便使得應用程式能夠自安全及隱藏分割讀取FAT。分割持續開啟直至主機終止會話或裝置經歷功率循環為止。只要分割開啟，便允許由主機應用程式讀取及寫入至隱藏分割。

圖11B說明安全分割中之檔案之主機讀取。如上文所描述，特定應用程式讀取命令出現於兩個卡協定命令內：指定讀取命令之性質及自變數的寫入命令；及接著的實際讀取命令。

圖11C說明至隱藏分割之主機寫入。在通過鑑認的情況下，主機可存取分割之FAT且現具有將新檔案傳送至主機之權限。主機將寫入命令發送至具有一寫入開始位址及區段計數的安全分割。

圖8展示上文提及之裝置應用程式狀態以及隱藏分割實例之例示性值的實施例。因為可允許若干不同分割，所以

在實例中裝置應用程式分割ID中之位元組1係給定為"1"。接著四個位元組給出分割大小，此處展示為區段之十兆位元組值。位元組5指示分割當前是否為開啟的且給定接著的四個位元組以指示裝置應用程式之版本。位元組10提供操作狀態。位元組10至511係以零進行填充從而形成一區段。在變化中，位元組11可用於記錄在最末異動中所寫入的區段之數目。

表1描述發送至隱藏分割實例之裝置應用程式的應用程式協定命令之一實例。CMD索引置放於命令區段之位元組33中，且命令自變數係自位元組34開始置放並以表中所寫入之指定次序而置放。

CMD索引	CMD名稱	CMD自變數	位元組之數目
1.	OPEN_PARTITION	以位元組之使用者名稱長度	1
		使用者名稱	指定於第一自變數中
		以位元組之使用者密碼長度	1
		使用者密碼	指定於第三自變數中
2.	CLOSE_PARTITION	以位元組之使用者名稱長度	1
		使用者名稱	第一自變數中指定
		以位元組之使用者密碼長度	1
		使用者密碼	指定於第三自變數中
3.	READ_PARTITION	LBA數目	4
		讀取之區段計數	4
4.	WRITE_PARTITION	LBA數目	4
		寫入之區段計數	4
5.	READ_PARTITION_STATUS	分割ID	1

表 1

當藉由裝置之寫入協定(諸如SD協定)而傳送內嵌隱藏分割命令時，在建置實際係用於分割之裝置應用程式且並非

用於SD規則操作之寫入或讀取命令之後將藉由(例如)SD命令解譯器來調用分割的命令解譯器。在剖析命令區段之後，命令解譯器將調用適當例行程序。在一特定實施例中，應用程式之命令可具有作為第一自變數之參數，其指示MD命令是讀取命令還是寫入命令。第二參數可將指標傳遞至經配置之RAM中置放命令區段的位置。由於分割之裝置應用程式的相關資訊開始於區段之該點處，所以該指標將指向位元組33(命令操作碼索引)。可在第三參數中指定由此第二參數指向的緩衝器大小。

如已提及的，因為在剖析命令區段之後立即以與卡協定之彼等序列相同的序列來執行裝置應用程式之寫入命令，所以該等寫入命令為簡單的。舉例而言，在SD狀況下：

1. 接收SD寫入命令(1或多個區段)：在例示性實施例中第一區段總為命令區段。
2. 卡層剖析第一區段之第一32個位元組並識別命令區段之簽名。以指向命令區段之指標而調用裝置應用程式命令剖析器(解譯器)。
3. 在執行命令解譯器以識別內嵌應用程式命令之後，命令解譯器調用將執行內嵌命令並開始應用程式之命令過程的相應例行程序。
4. 執行隱藏分割之應用程式的命令過程且將狀態傳回卡層，其中序列將轉至如SD寫入命令過程將終止的結束處。SD卡轉至閒置(IDLE)狀態。

亦如已提及的，隱藏分割之應用程式協定中的讀取命令

為兩步驟過程，其包含(再次在SD實施例中)SD寫入命令，隨後為SD讀取命令。第一步驟為SD寫入命令：

1. 由主機發送的具有一資料塊(512個位元組)之SD寫入命令：在例示性實施例中第一區段總為命令區段。
2. 卡層剖析第一區段之第一32個位元組並識別命令區段之簽名。以一指向前命令區段之指標調用裝置應用程式之命令剖析器(解譯器)。
3. 在執行命令解譯器以識別內嵌應用程式命令之後，命令解譯器識別將(在接著的SD讀取命令中)被執行的內嵌讀取命令，儲存此命令識別符。接著命令解譯器設定卡層讀取命令旗標。
4. 執行應用程式之命令過程且將狀態傳回卡層，其中序列將轉至如SD寫入命令過程將終止的結束處。SD卡可轉至閒置狀態。

第二步驟為SD讀取命令：

1. SD讀取命令。
2. SD卡層檢查應用程式卡層讀取命令旗標並發現其被設定：其調用具有讀取命令指示的裝置應用程式命令解譯器而非調用規則SD讀取命令。
3. 裝置應用程式命令解譯器識別分割讀取命令旗標並重新調用儲存於先前寫入命令中的分割讀取命令識別符。該識別符指向相應分割讀取例行程序並初始化分割讀取序列。
4. 執行分割讀取序列並將傳回狀態傳遞至卡層。

5. 卡層終止SD讀取過程且卡進入閒置狀態。

藉由基於SD之實施例之圖12中所示的卡側應用程式之命令狀態機而說明讀取及寫入過程。由於此等內嵌命令係藉由SD寫入命令而傳遞，且當需要執行隱藏分割讀取命令時主機不能夠傳遞SD讀取命令上的命令區段，所以裝置應用程式(1023)之命令解譯器需要保持一將管理及瞭解哪一讀取命令正由主機應用程式1053執行的狀態機。在此圖中，圓圈係指執行於裝置應用程式層(圖6，1023)處的裝置特定應用程式命令，同時附加註釋係指如在FE層(圖6，1021)處所見的在中間(此處，SD)協定中之命令。圖12之右側(A路徑)為寫入處理之過程，其中將資料傳送至裝置或無資料被傳送的任何過程將遵循類似之流程。圖12之左側對應於其中將資料自裝置傳送至主機的讀取或其他過程。讀取流程由讀取所需之該對之第一命令的B路徑及該對之第二命令的C路徑組成。

寫入流程與讀取流程以相同之方式開始，接收一SD寫入命令，在偵測到簽名之後自該SD寫入命令而剖析內嵌命令。對於寫入流程而言，寫入旗標保持為開(on)且執行寫入過程。然而，對於讀取流程之第一階段而言，將設定讀取旗標。當偵測到來自主機之讀取過程時，藉由裝置應用程式(1023)而設定此卡層(SD)讀取過程旗標。該旗標向卡層指示來自主機(在SD匯流排上)之下一讀取命令並非規則卡讀取且應將其導引至裝置應用程式(1023)以供處理。此發生在讀取流程之第2階段，其中回應於該對之第二命令而自裝置

傳送資料。當卡層初始化時旗標獲得重設。

參看表1中所列之命令，OPEN_PARTITION命令(OP-Code 0x01)接收使用者名稱及密碼並對其進行驗證。若其與裝置應用程式之所儲存名稱及密碼相匹配，則開啟分割以用於主機存取(讀取、寫入)。此時，存在著將被發送及相對於裝置應用程式之碼中的片語而被驗證的鑑認片語。圖13A中展示此過程。分割將保持開啟直至CLOSE_PARTITION將其關閉為止或直至下一功率循環為止。

CLOSE_PARTITION命令(OP-Code 0x02)之使用係用以關閉應用程式工作會話且不讓分割開啟。圖13B中展示一實施例。

圖13C中展示READ_PARTITION命令(OP-Code 0x03)之使用。若分割開啟(自1633至1635為是(yes)路徑)，則此命令將使得主機應用程式能夠讀取儲存於分割中之資料(1639、1641)。在例示性實施例中，此例行程序將調用"FlashRead"API例行程序，其將資料傳送導引至隱藏分割空間。

WRITE_PARTITION命令(OP-Code 0x04)將資料自主機導引至卡上之隱藏分割。如圖13D中所示，僅當設定(SET)分割旗標(1653)時，此命令才由裝置應用程式加以執行。此例行程序將調用"FlashWrite"(1659、1661)API例行程序，其將資料傳送導引至隱藏分割空間。

READ_PARTITION_STATUS命令(OP-Code 0x05)供應圖8中所示之資訊。此為具有上文參看圖8所述之結構的應用程

式讀取命令。當命令進入時，例行程序將聚集所有資訊且將其置放於上述結構中並將其發送至主機。

如上文所提及的，某些作業系統將不時地將一指令分開，使得並非所有資料部分將附著至其相應命令部分。在此狀況下，寫入指令之資料部分將在無卡傳遞簽名的情況下顯露於卡處；因此，表現為不具有附著簽名之傳輸協定中之標準命令的命令可實際上為剩餘特定應用程式命令。作為具體實例，考慮圖5之指令510的狀況，指令510具有(例如)十個區塊之資料部分513。在將此指令傳輸至卡時，主機之作業系統可僅發送附著至該指令之資料部分的首選五個區塊，其中稍後傳輸剩餘五個區塊。當發送此最末組之資料塊時，其將缺乏含有應用程式協定之簽名及實際命令的初始資料塊513a，且將表現為來自傳輸協定之標準命令的部分。當內嵌特定應用程式命令為讀取型命令時，此分段亦可發生，使得以部分的方式將資料自卡發送至主機。圖10B藉由以用於讀取及寫入過程之邏輯位址 LBA_{Ri} 及 LBA_{Wi} 替代單一邏輯塊位址 $LBA=LBA_XYZ$ 來擴展圖10A之流程從而說明在特定應用程式讀取與寫入命令中的此分段。藉由界定 LBA_{Ri} 讀取以及寫入之 LBA_{Wi} ，裝置可處理在任一方之應用程式資料傳送的中斷。下標 i 係用以慮及來自多個應用程式及多使用者主機裝置之多個特定應用程式命令之同時發生的實施。

圖10B之過程再次基於將SD協定用作傳輸協定的SD卡之實例並以處於SD閒置狀態及在1201處接收命令的裝置而開

始。其接著判定所接收之指令是否為SD寫入(1203)，且若並非SD寫入，則判定其是否為SD讀取(1205)。若其既非SD寫入亦非SD讀取，則執行SD命令(1207)且裝置返回至閒置狀態(1209)。

若在步驟1203處所接收之指令為SD寫入，則接著檢查傳遞簽名(1214)。若在1214處發現簽名，則此指示新內嵌命令之開始，該新內嵌命令可接著被提取。在1231處，對於所有讀取及寫入邏輯塊位址 $LBA_{Ri;Wi}$ 而言，若傳入命令之邏輯塊位址 LBA_{CMD} 與指派給當前寫入過程的邏輯塊位址相同($LBA_{CMD}=LBA_{Wi}$)，則清除相應 $LBA_{Ri;Wi}$ 。通常，所指派之 LBA_{CMD} 應未被使用，但在其已被指派給另一過程的狀況下，此會將其異常中止且清除位址。

接著判定特定應用程式命令之資料方向(1233)。若不涉及資料傳送，則接著執行特定應用程式命令(1245)且裝置返回至閒置狀態(1249)。

若應用程式之命令為寫入，則將相應邏輯位址設定為 $LBA_{Wi}=LBA_{CMD} + \text{data blockcount}$ (1251)且寫入應用程式資料(1253)。如上文所論述的，主機之OS可將指令分開，使得命令不隨附有所有相應資料。當提取及剖析內嵌命令時，卡將瞭解，應包括多少資料塊。在1255處，進行檢查以察看是否包括並寫入所有資料。若如此，則可清除 LBA_{Wi} (1257)；否則，其不被清除，因為剩餘資料將(除非由錯誤或停機而中斷)最終跟隨且將看似為至一作為上文所計算之位址的連續(相對於此命令)卡位址之標準SD寫入

命令。裝置接著將返回至閒置狀態(1259)以等待另外的命令。

返回 1231，若應用程式命令為讀取，則裝置設定 $LBA_{Ri}=LBA_{CMD}$ 並設定 $datatype_{Ri}=datatype_{CMD}(1235)$ 從而為在 1263 處於第二讀取階段中的隨後資料傳送作準備。裝置接著轉至 SD 閒置模式以等待讀取過程之第二階段。對此解決方案之不同的變化將不使用 LBA_{CMD} 而是使用在寫入命令之參數中明確界定的不同位址 LBA_{READ} 。此方法將圍繞 OS 之快取機制 (caching mechanism) 而工作。在若干系統中，主機 OS 將不會試圖往回讀取剛剛被寫入之卡位址。實情為，其將會將資料緩衝器返回至主機應用程式之主機快取緩衝器 (cache buffer)，假定此資料為卡所具有之資料，由於其剛剛被寫入至其。在此狀況下，資料緩衝器將包括特定應用程式命令而非卡回應。

返回至 1214，若命令(在傳輸協定中)為寫入(自 1203 為是 (yes) 路徑)，但缺乏簽名(自 1214 為否 (no) 路徑)，則該命令實際上可為傳輸(此處為 SD)協定中之寫入命令，或可為用於應用程式協定中之寫入的額外資料。在 1215 處對此進行判定：若對於 LBA_{Wi} 中任一者而言， $LBA_{CMD} \neq LBA_{Wi}$ ，則其為接著被執行於 1217 處的標準 SD 寫入，其後卡轉至 SD 閒置狀態(1219)。若 LBA_{CMD} 匹配 LBA_{Wi} 中之一者，則命令實際上為相應應用程式命令之更多資料部分且流程轉至 1251 以寫入此額外應用程式資料。

返回至步驟 1205，若命令為 SD 讀取，則判定其實際上為

SD讀取或為應用程式讀取之第二階段。在1226處藉由比較 LBA_{CMD} 與 LBA_{Ri} 而對此進行判定：若對於 LBA_{Ri} 中任一者而言， $LBA_{CMD} \neq LBA_{Ri}$ ，則其為接著被執行於1227處的標準SD讀取，其後卡轉至SD閒置狀態(1229)。若替代地，對於 LBA_{Ri} 中任一者而言， LBA_{CMD} 匹配 LBA_{Ri} 中的一者，則其為應用程式讀取之第二階段。

當 $LBA_{CMD} = LBA_{Ri}$ 時，將相應邏輯位址設定為 $LBA_{Ri} = LBA_{CMD} + \text{data blockcount}(1261)$ 且讀出應用程式資料(1263)。裝置接著轉入SD閒置狀態(1269)。對於應用程式讀取過程，例示性流程缺乏應用程式寫入側之步驟1255及1257的均等步驟。即使已存取所有相應資料，亦繼續設定相應讀取旗標。藉由保持 LBA_{Ri} 開啟，可在錯誤狀況下重新讀取資料。在此配置下，對於一實例，諸如在步驟1231中，若為另一目的而重新指派位址，則僅清除 LBA_{Ri} 。

相對於圖10B之圖10A的此等添加允許在應用程式資料之傳送中存在中斷。其亦消除了對讀取及寫入至相同邏輯位址之需要。此等允許技術用於其中可能不存在對作業系統具有足夠控制的狀況下，因為圖10A之實施例所需的控制之種類可能需要已知可控平臺。同樣，應再次注意：圖10B更明確地合併可存在多個主機應用程式，其各自與卡進行獨立通信，且資料傳送之中斷可在來自一協定之指令的部分之間插入來自另一協定之指令的部分。另外，讀取過程與寫入過程之獨立LBA允許繞過裝置驅動程式層中之任何讀取或寫入快取。

第二例示性實施例

第一實例實施於檔案層級。本發明之各種態樣亦可實施於裝置驅動程式層級，從而藉由將請求發送至裝置驅動程式而初始化讀取及寫入操作，例如在主機使用 Windows 作業系統時該裝置驅動程式可為 Windows 標準儲存驅動程式。命令傳遞方法之此實施直接與裝置通信且不具有檔案系統額外耗用。然而，此方法可需要管理權限。主機之應用程式可嘗試此方法並將此方法用作第一選擇且若其失敗則轉移至檔案系統操作。介面功能可用於在該等方法之間進行選擇並相應地初始化通信管線。在任何其他操作之前，用戶端應用程式將調用此命令。若應用程式缺乏適當管理員權限，則以此方法工作將導致異常，其可根據異常處理碼而被處理。

此實例係基於對上文[先前技術]段中問題之論述。儘管[先前技術]之論述提出在記憶卡經由硬體卡讀取器而連接至PC之情形中的問題，但將在下文中看到(如同此處給出之其他實例一般)，情況更為一般。記憶體裝置可具有此協定轉譯，或更精確言之，即使當其直接連接至第二主機時亦無轉譯問題。當記憶體裝置連接至具有(如自記憶體裝置所看到的)不完整協定之主機時，可導致此情況。一其中可出現此的狀況為其中卡具有兩組接點，一組供USB埠使用且另一組供一標準組之卡接點使用。(在皆於2004年4月16日申請之美國專利申請案第29/203,693、US 10/826,801及US 10/826,796號中描述此等雙接點卡之實例。)如關於第三組

實施例所論述的，協定轉譯可完全地包含於記憶體裝置自身內，不為主機所知。此外，在下文中，儘管關於可拆卸記憶卡而描述實施例，但該論述亦適用於內嵌記憶體裝置。

在第二例示性實施例之第一組實施例中，為允許媒體卡特定命令自主機傳遞至卡而不改變控制媒體卡之配接器或讀取器的韌體，本發明使用特定邏輯塊位址(LBA)(卡傳遞模式LBA)，其具有在資料區段中之特定簽名以通知卡特定命令內嵌於區段中。此可藉由韌體變化來實施以支援命令傳遞(CPT)模式。因為此不需要配接器之韌體變化，所以可在任何USB讀取器或配接器處執行該卡。

如在[先前技術]之段中所論述的，各種小型快閃記憶卡(緊密快閃卡、安全數位(SD)卡、多媒體卡(MMC)、xD及索尼記憶棒/高速記憶棒等)具有不同電氣介面及通常供主機(數位相機、快閃記憶體音樂播放器)使用的特定媒體命令，該等特定命令未在PC與允許PC讀取卡的硬體配接器之間所使用的命令集中發現。

在一主要態樣中，本發明使用一卡韌體變化以執行正常讀取命令/寫入命令。然而，當至特定LBA之讀取命令/寫入命令偏移時，檢查簽名以判定命令之資料部分是否含有內嵌媒體卡特定命令。可將協定實施於任何媒體卡之韌體中。因此，不需要配接器、USB讀取器或由PC(或不可存取完全特定媒體命令集之其他主機)使用的其他讀取器之任何韌體變化。實際上，對於卡製造者而言，此比實施讀取器韌體改變以用於所有讀取器廠商中的所有卡命令簡單。

儘管通常將根據特定卡(諸如安全數位(SD)或緊密快閃(CF))及特定協定(諸如SCSI或ATA)而描述本發明之各種態樣，但當論述實例時，將瞭解，各種態樣適用於其他各種記憶體及系統。在整個此文獻中以引用方式併入的以下專利及申請案中進一步描述關於卡結構及操作之各種細節。

圖14展示卡與個人電腦(或更一般而言，需要硬體配接器以讀取該卡的任何主機)使用讀取器進行通信所用之過程。圖14除忽略對在某些狀況下的USB包裝或類似結構的明確介紹(其在以下論述中亦將被限制)之外，其類似於圖4。如[先前技術]中所提及的，卡與讀取器使用來自卡協定之命令集的(諸如201處所指示的)指令而通信。主機與讀取器通常使用來自另一命令集的(諸如203處所指示的)指令而通信。讀取器使用其韌體而在兩個命令集之間轉譯。(儘管此處及下文中提及韌體，但更一般而言此可實施於硬體、軟體或此等硬體與軟體之某組合中。)困難在於當命令201為來自在由PC用以與讀取器通信之協定中不具有均等命令的卡之命令集之命令時。在此狀況下，因為不存在此均等指令203，所以讀取器不能夠將201轉譯成相應命令203，且對於相反方向而言亦為類似的。因此，若主機希望使用卡之特定媒體指令中之一者，則在不改變PC硬體配接器協定之命令集的情況下於先前技術中沒有辦法經由讀取器傳遞該指令。

此特定實例將使用自主機至讀取器之SCSI協定、自讀取器至卡之SD協定及不會作為內嵌命令存在於SCSI協定中

的SD命令(例如，安全寫入)之實例。所以儘管此命令可直接在SD卡中傳輸，但其需要使用應用程式之主機側的本發明態樣以能夠傳輸載運主機與讀取器之間的命令之中間SCSI協定中之指令。在讀取器處，虛設命令在每一協定中具有一版本且可在該等協定之間被轉譯，其中將實際內嵌命令視為資料。舉例而言，虛設命令再次被視為寫入。因此，在主機與配接器之間，其將表現為SCSI寫入命令且在配接器與卡之間其將表現為SD命令，同時在該兩個狀況下，實際命令內嵌於資料部分中。所以儘管在此實例中內嵌命令現存在於第二協定中，但其仍為內嵌的，因為在沿路徑之某一點處，其傳遞一缺乏此特定應用程式命令之協定。因此，此導致第一協定之命令內嵌於相同協定之命令中，從而允許在其與缺乏內嵌命令之均等命令的另一第二協定之間進行轉譯。圖15中對此進行了展示。儘管此論述主要係根據SCSI及SD實施例而論述，但將瞭解，一般而言其適用於其他協定。

圖15為當來自不具有PC讀取器命令集中之均等命令之卡命令集的命令內嵌於具有在兩個命令集中之表示的PC讀取器協定之指令203中時，本發明之原理態樣的示意性表示。在例示性實施例中，特定媒體命令再次內嵌於PC讀取器協定之資料部分中。PC及讀取器可接著交換命令203，其中讀取器可將該命令203轉譯為相應命令201。然而，在此命令內內嵌有另一特定媒體命令601。讀取器接著傳遞此命令並以一對其操作透明之方式轉譯該命令。例示性實施例之機

制為具有在兩個命令集中之表示的指令201/203可存取卡之記憶體中的預定邏輯位址。當(例如)卡接收到至特定邏輯位址之讀取命令時，警告該卡此讀取命令可為特定媒體命令601。(如下文中所描述，下文之實施例不依賴於此特定邏輯位址，且更一般而言，不僅可使用寫入命令而且可使用具有資料部分之任何命令。)卡接著檢查此命令之簽名以對此進行驗證且接著提取命令601。在例示性實施例中，由於在SCSI及其他協定中，通常不存在方便之空間將特定媒體命令內嵌於指令之命令部分中，所以命令601置放於指令之資料部分內。

舉例而言，若PC希望對SD卡之安全區域進行寫入，則其將形成指令203，該指令203由至特定邏輯塊位址(LBA)之寫入命令組成，含有命令傳遞(CPT)簽名並含有內嵌於資料部分中之實際安全寫入命令601。PC接著將此指令203發送至讀取器，該讀取器將其解譯為SCSI協定中之標準寫入指令。接著藉由讀取器將此轉譯為SD協定中之標準寫入指令201。讀取器恰好傳遞安全寫入命令601，假定其為附著至寫入命令之資料的部分。當指令201到達卡時，基於為寫入而指定的LBA，韌體允許卡判定指令替代地係為特定媒體命令。控制器接著提取命令601(將其識別為安全寫入)，且藉由執行對指令之實際資料部分的安全寫入而繼續執行命令。

圖16更詳細地描述執行至SD卡之N個資料區段之安全資料寫入的實例之一特定實施例。頂列展示如由主機形成之

指令。因為安全資料寫入不具有實例之SCSI協定中的類似物，所以命令部分由SCSI協定中之命令CMD'組成，CMD'此處為寫入，其存在於兩個命令集中。寫入係至特定命令傳遞邏輯塊位址CPTLBA。(未展示命令之其他細節，以簡化論述。)資料部分(虛線之右邊)由初始區段組成，其含有特定媒體命令，隨後為N個實際資料區段。初始資料區段將包括命令傳遞簽名、實際特定媒體命令及其中將寫入以下資料的實際位址。邏輯塊位址CPTLBA對應於上文論述之第一例示性實施例的論述中所使用之邏輯塊位址LBA_XYZ(例如見圖10A。)不同標記係用於指示在實施於主機側裝置驅動程式層級的本實例中，正被使用之邏輯塊位址為特別指定之邏輯塊位址，而在實施於檔案層級之先前實例中，可將其視為任何位址。

圖16類似於先前論述之圖5，但其中對於在此實例中明確展示之傳輸協定(主機至讀取器、讀取器至裝置)中每一者重複一次指令510。圖5之指令510將命令CMD之內嵌展示為此將發生於沿應用程式之主機側與裝置側之間的傳輸路徑之某處，同時指令530將所提取之形式展示為其將由應用程式之裝置側使用。圖16說明可存在若干中間協定且實際命令(CMD)將保持內嵌於若干此等中間協定中，即使在其中內嵌命令具有特定中間協定中之類似物的狀況下亦為如此。

如由讀取器所見，第一資料區段恰好為其視為待寫入至特定邏輯塊位址之資料的N+1個區段中之第一者。同樣，此等傳遞內容未改變。藉由讀取器將指令之命令部分自PC/

讀取器協定中之其表示(CMD')轉譯為卡協定中之其表示(CMD)。此時，可將指令傳遞至卡且該指令仍具有(例如)至CPTLAB之寫入命令隨後為N+1個資料區段的形式。

一旦在卡中，韌體便可解開實際命令。當其判定命令使用特定位址時，其接著轉至第一資料區段，檢查CPT簽名並提取實際特定媒體命令。此接著形成指令之實際命令，該實際命令之後將接著為N個實際資料區段。

因此，如參看圖16描述的，具有N個資料區段之特定媒體寫入命令變得內嵌為在於PC與讀取器之間使用的傳輸協定中之N+1個資料區段。為了讀取N個資料區段，將特定媒體讀取命令作為在讀取(或接收資料之其他命令)或於PC與讀取器之間使用的傳輸協定中之一資料區段而內嵌。因此，可將特定媒體讀取命令實施為具有內嵌於僅資料區段中之特定媒體讀取命令的寫入命令。再次基於特定邏輯位址(CPTLBA)及簽名，卡將提取特定媒體命令並以來自特定媒體讀取中所指定的經請求邏輯塊位址之資料進行回應。上文參考第一例示性實施例進一步展開關於讀取實施之細節及處理諸如寫入或讀取失敗之複雜問題的方法。

特定邏輯位址(CPTLBA)較佳為並非通常用於檔案系統中之區段，因為此可避免與可被發送至此區段之標準讀取命令或寫入命令相衝突。舉例而言，在DOS中，於主啟動記錄之後，通常存在通常不被存取之某些隱藏區段。(在某些作業系統中，存取此區域可能需要管理員權限，其為在下一段之基於檔案之實施中可繞過的可能之複雜問題。)藉由

將此邏輯位址視為特定邏輯位址，若非出於檢查實際命令為內嵌特定媒體命令之目的，對此位址之存取將不會真正在此處讀取或寫入資料。讀取器將僅傳遞此命令作為正常資料存取且卡將每件事分類。

圖 17A 至 17L 展示可如何將命令之各種實例內嵌於資料區段中的一實施例。如上文所描述，當執行至媒體卡之寫入時，資料部分之第一區段含有 CPT(卡傳遞)命令。其指示待執行之 CPT 功能。因此，為了寫入 10 個區段，因為第一區段為 CPT 標頭故實際上寫入 11 個區段。為讀取 10 個區段，使用者需要發送 CPT 命令之一寫入接著讀取 10 個區段。實際應用程式為執行不可由正常讀取命令或寫入命令達成的特定命令。舉例而言，CPT 模式可用於特定媒體命令，諸如 AKE(鑑認密鑰交換)或 SD 卡之應用程式命令。

圖 17A 展示內嵌命令格式實施例。位元組 0 至 31 中之 CPT 簽名為取決於用以檢查該簽名與該命令匹配之命令的字符串。位元組 32 中之命令可包括：

0 - CPT 模式檢查：意欲檢查媒體卡之 CPT 模式支援

簽名 "卡傳遞模式檢查"

1 - 資料輸出：資料將寫入至媒體卡

簽名 "卡傳遞輸出命令"

2 - 資料輸入：資料將自媒體卡讀取

簽名 "卡傳遞輸入命令"

3 - 無資料傳送之命令

簽名 "卡傳遞無 I/O 命令"

4 - CPT命令異常中止

簽名"卡傳遞異常中止命令"

5 - CPT模式重設

簽名"卡傳遞重設命令"

6 - CPT狀態擷取

簽名"卡傳遞狀態讀取"

資料傳輸長度(Data Transfer Length)(位元組33至35)指定待傳送之資料量(如位元組之數目);由於資料以區段形式傳送,所以不完整之區段由0來填充。Data(位元組36,位元1)為資料傳送之旗標且若在下一命令上不存在資料傳送則其為"0",若在下一命令上存在資料傳送則其為"1"。類似地,Dir(位元組36,位元0)為資料傳送之方向的旗標且若資料傳送係對於下一命令上之輸入而言,則其為"0",若資料傳送係對於下一命令上之輸出而言,則其為"1"。

媒體卡特定命令區域(位元組48-511)為當被提取時媒體卡將對其加以執行的實際命令。媒體卡特定旗標區域(位元組36,位元0)用於媒體相關旗標。

若在發送實際命令之前媒體卡支援卡傳遞命令協定,則必須檢查該媒體卡。如上文所描述,LBA(CPTLBA-卡傳遞LBA)係指定用於該協定。LBA可為自卡之低0區段至最末區段的任何數字。在實例中,此將為LBA 2,其為在FAT檔案系統中之主啟動記錄之後的第二隱藏區段且通常並不用於保持資料。注意,即使該LBA含有有效資料,此協定仍將保存其值。對於檢查模式支援,存在兩個選擇。

查詢協定支援之第一選擇係用於當媒體卡將CPT模式支援簽名置於CPTLBA區段中時。然而，若CPTLBA為具有有效資訊之區段，則此並非較佳的；例如，FAT/目錄或使用者資料區域中之區段。若支援CPT模式，則視情況卡可返回具有如圖17B中所示之簽名的CPTLBA區段，其中位元組0-31為"支援卡傳遞模式(以空白填充)"且位元組32-511為0。

若CPTLBA區段不含有支援CPT模式之簽名，則可使用查詢協定支援的第二選擇。在此狀況下，將以下協定用於檢查模式支援：

- a)讀取CPTLBA區段並將其儲存於主機記憶體區域中。
(不支援簽名)
- b)寫入具有一含有用於查詢CPT模式支援之簽名之區段的CPTLAB。
- c)讀取CPTLBA區段以檢查CPT模式支援之回應。
- d)寫入具有一來自所儲存記憶體區域之原始區段的CPTLAB。

當主機讀取CPTLBA區段並將其儲存於記憶體中時，步驟a)允許檢查特定邏輯位址(CPTLBA)以進行支援同時仍保持其中可含有之任何資料。此係用以在媒體卡不支援卡傳遞模式時復原區段。此寫入並不使用CPT模式命令。

圖17C展示步驟b)，具有含用於查詢之簽名之區段的CPTLBA寫入。主機將CPTLBA區段的位元組0-31寫入為"卡傳遞模式檢查"(以空白填充)，接著於位元組32-47中寫入命

令及旗標。位元組48-511為0。

在步驟c)中，主機讀取CPTLBA區段以進行回應。若回應如圖17D中所示具有為"支援卡傳遞模式"(以空白填充)之位元組0-31及為0之位元組32-511，則支援卡傳遞模式；否則，卡並不支援CPT協定。注意，此協定不同於用以查詢協定支援之第一選擇。當發出CPT模式檢查命令時，對簽名進行回應。

在步驟d)中，主機自所儲存之記憶體區域寫入CPTLBA區段中的原始資料。注意，即使卡支援卡傳遞模式，仍因為不存在適當簽名而不會將此寫入解釋為特定CPT命令。

參看圖17E至圖17G而描述輸入/輸出協定。為將資料輸出至媒體卡，首先以用於輸出之簽名寫入CPTLBA，接著基於資料傳送長度而寫入資料。為了以用於輸出之簽名寫入CPTLBA，主機將一寫入命令發送至媒體卡，其中資料具有內嵌有媒體卡特定命令的CPT命令格式。長度為"資料傳送長度"加CPT命令格式之一區段。圖17E中對此進行展示，其中位元組32中之1對應於"待寫入至媒體卡的資料"。

圖17F中展示基於資料傳送長度之實際寫入，其中媒體卡如何處理資料取決於媒體卡特定命令。如參看圖17A所提及的，此實例中之"資料傳送長度"資訊係根據位元組而表示於位元組33-35(MSB、LMSB、LSB)中。然而，因為在卡與讀取器之間以區段形式傳送讀取/寫入資料，所以可視需要以0來填充不完整區段。舉例而言，在SD卡中，鑑認密鑰交換(AKE)過程使用8個位元組進行詢問及回應。因此，主機

以504個位元組的0來填充資料區段以詢問1及回應2資料，其中卡類似地填充詢問2及回應1資料。

為了輸入來自媒體卡之資料，主機以用於輸入之簽名寫入CPTLBA且接著基於資料傳送長度而讀取資料。圖17G展示待發送以指示將讀取資料的CPT命令，其中位元組32中之2對應於"待自媒體卡讀取的資料"。為了基於資料傳送長度而讀取資料，於主機將一讀取命令發送至媒體卡之後，待讀取之區段的數目為 $[(\text{資料傳送長度})/512]$ 。(此假定每區段512個位元組；更一般而言，若為不同區段，則實際區段長度將取代512。)起始區段為CPTLBA。由於先前命令為CPT輸入命令，所以卡會基於媒體卡特定命令而將資料發送至主機。

不具有輸入或輸出之命令的實例包括"不具有資料傳送之命令"、"CPT命令異常中止"及"CPT模式重設"。圖17H展示"不具有資料傳送之命令"，如由在位元組32處之3所指示。

圖17I展示"狀態讀取"命令(由位元組32處之6指示)，其允許主機讀取CPT狀態。舉例而言，在SD卡中，可將狀態實施為回應資料。該狀態將為讀回資料之最初少數個位元組。

儘管當需要具體實例時參考SD卡，但圖17A至圖17I並不限於任何特定媒體協定且並不限於特定媒體命令。為了給出特定媒體命令之實例，圖17J展示專用於SD或MMC卡之內嵌命令且圖17L展示專用於緊密快閃卡之內嵌命令。

在圖17J中，展示SD/MMC實例之命令格式。位元組0-35

係如上文所述。SD/MMC命令填充媒體卡特定命令欄位。位元組48-53為SD命令，例如為安全讀取。在媒體卡特定旗標欄位中需要並界定某些旗標。BLKH指示命令是用於單一區段還是用於多個區段：若使用單一區段命令則為0，且若使用多個區段命令則為1。對於應用程式命令而言，APP=1。回應類型視命令索引而為SD回應類型。位元組48-52中之各項為SD命令之實際元素，其中位元組53中之CRC7視情況可被設定為0。

為獲得SD命令之回應資料，使用者可發送狀態擷取以獲得回應。圖17K中對此進行展示。當讀取區段之最初6個位元組時將傳回回應資料。

圖17L展示內嵌命令之緊密快閃(CF)命令格式。在此狀況下，命令在位元組48-54中且提供包含CF命令所需之元素。在圖17L中，媒體卡特定旗標(位元組36之位元2至位元組37)經保留以供專用於各種卡格式之另外命令之用。

第三例示性實施例

第三例示性實施例實施於檔案層級，如第一例示性實施例一般。此處，論述將更集中於主機側檔案中所置放之細節。由於裝置特定命令內嵌於檔案中，所以沒有辦法損害諸如FAT區域之檔案系統特定資料。自卡之觀點而言，不同例示性實施例將看起來為類似的，其不同之處更多的是關於如何在主機側封裝資訊。當需要參考特定應用程式時，第三例示性實施例將為至安全匯流排之安全連結的應用程式，諸如在2004年12月21日申請的美國臨時專利申請案第

60/638,804號中所提出的，該案以上述方式併入本文中。

在使用檔案層級實施的情況下，除克服使用裝置驅動程式層級實施可能出現的該類權限問題之外，亦可克服對具有發送特定命令之特定硬體的可能之需要。舉例而言，在某些狀況下，於裝置驅動程式層級，存在著必須發送特定指定以存取SD卡之安全區域的需要，但在使用檔案層級實施的情況下，因為資訊係封裝於檔案內，故對此不再存在需要。所以，只要可寫入及讀取檔案，便可以此實施來發送及接收任何命令集。

先前段之實施例為卡驅動程式實施，其操作於裝置輸入/輸出層級。主機或PC應用程式與裝置或卡驅動程式之間為作業系統(OS)檔案系統。第三組實施例為檔案I/O實施，其對OS檔案系統進行操作。在此等實施例中，主機僅告知檔案系統將一檔案寫入至記憶體裝置，其中再次內嵌裝置特定命令。關於將檔案寫入至快閃記憶體的更多細節係描述於皆在2005年2月16日申請之美國專利申請案第11/060,249號、第11/060,174號及第11/060,248號中，該等案以引用的方式全部併入本文中。儘管此等實施例可需要比裝置層級實施例多一些的韌體額外耗用，但其可具有若干優點。舉例而言，在裝置層級，當開發應用程式時，需要瞭解一些有關硬體，而藉由使用檔案系統，此變得與硬體無關並與底部層級協定之許多細節無關。

如同裝置層級實施例一般，檔案層級實施例可用於將廠商特定命令發送至各別廠商產品，諸如在 Windows

98/ME/95/DOS及類似實例中所建置的進階SCSI程式介面(ASPI)。在上文描述之裝置層級實施例中，使用至特定邏輯位址(CPTLBA)之寫入。然而，在許多作業系統中，為存取此邏輯位址需要某些存取權限。舉例而言，若使用者不具有管理員權限，則不能夠將某SCSI命令發送至廠商產品。因為作業系統通常允許寫入檔案，所以本發明之檔案層級實施例藉由將特定媒體命令內嵌於檔案中、將檔案寫入至裝置及在裝置上解除內嵌實際命令而防止權限問題的發生。在Windows作業系統之許多版本中，使用例示性CPTLBA之卡層級實施例將需要管理員權限，同時寫入一檔案將不需要任何此等特定存取權限。如前所述，例示性實施例將特定媒體命令內嵌於指令之資料部分中。

在作業系統之檔案系統層級處的實施可再次由卡之卡韌體加以實施，該卡之卡韌體經調適以遵守並識別協定。使用至任何LBA之讀取命令/寫入命令，檢查簽名以判定資料是否含有內嵌之媒體卡特定命令。此不同於其中使用特定位址之裝置驅動程式實施。在其他方面，如自卡所見的，在此段之裝置驅動程式層級實施例與先前段之檔案系統層級實施例之間存在很少差別。

如在先前段中，若命令涉及任何資料傳送，則為了保持與命令相關聯之資料，基於檔案之實施的例示性實施例將協定分成"指令"部分(其中內嵌有特定媒體命令)及"資料輸入/輸出"部分。當由卡接收到協定命令時，在讀取或寫入操作期間韌體檢查特定簽名以判定給定LBA是否為指令部

分，此僅需要對其中簽名總內嵌於第一資料區段中之實施例之協定命令的第一資料區段加以執行。自主機側程式員之觀點而言，用戶端應用程式發出簡單檔案或具有指令部分及(也許具有)資料輸入/輸出部分的邏輯區段(LBA)寫入操作。若命令涉及讀取資料或寫入資料至媒體，則韌體執行邏輯至實體映射。韌體將在自傳輸協定接收到作為寫入或讀取之物期間偵測API指令部分。

在例示性實施例中，協定命令之指令部分將含有簽名、查詢命令、廠商特定命令及可更新欄位。此處大小將限於512個位元組(其為最通常使用之資料塊的大小)之倍數。圖18A中展示例示性實施例中之指令頁的格式，其除了具有額外細節之外極類似於圖17A。首先36個位元組未經加密，其中剩餘部分係根據陳述經支援加密演算法之存在的加密/解密資訊而加密/解密。

關於圖18A中所示之各種欄位，位元組0及1為用於指示一可為API指令頁之候選者之LBA的簽名位元組。若此等兩個位元組匹配，則韌體將檢查API簽名。在以下實例中，位元組0為19，且位元組1為73。API簽名位於位元組2-34中且為單一字符串。在以下實例中，將簽名視為"進階程式介面"。

位元組35為告知韌體隨後指令頁及資料頁是否經加密的加密/解密(E/D)資訊。若其未經加密，則應將此欄位設定為API_NO_ENCRYPTION；否則，應將其設定為加密名稱，使得韌體可解密及加密自位元組36開始的指令頁或多個指令頁以及資料頁。位元組36-39為韌體操作狀態欄位

(Operation Status Field)欄位且將基於操作之成功或失敗而被更新。若一切皆好，則其將被設定為API_OP_SUCCESS；否則，其將被設定為錯誤值或API_OP_NO_SUCCESS。建議當發出命令時，將其預設值設定為0xFFFFFFFF。在寫入之後，調用者可讀取指令頁以驗證所請求之操作是否成功。可將廠商識別符欄位(位元組40-43)用作唯一廠商識別符。若韌體亦需要有效廠商ID，則此欄位可由調用環境用以識別韌體自身。

指令頁欄位(位元組44)之數目將指令頁之總數目告知韌體，其中指令頁以512個位元組為邊界。類似地，資料頁欄位(位元組45-46)之數目將附著至指令部分之資料輸入/輸出頁的總數目告知韌體。資料頁大小不必意味著以位元組之資料大小；在某些狀況下，可能已將填充添加至資料位元組之末端。舉例而言，DES需要以位元組之資料大小必須為八的倍數。位元組欄位(Bytes field)(位元組47-50)欄位中之資料大小將含有效資料之資料頁中的位元組之總數目告知韌體。在自媒體進行讀取之狀況下，此欄位應由韌體使用頁區域中實際處理之位元組大小加以更新。在將資料寫入至媒體時，此為資料大小之位元組的長度。若加密資料，則韌體首先解密資料頁之全部內容且接著進行提取。位元組51中之位元0為資料傳送之旗標，若當前命令不需要資料傳送則將其設定為"0"，且若當前命令需要資料傳送，則將其設定為1。

位元組51中之位元1及2為資料傳送之方向的旗標。值

"00"指示資料傳送係對於當前命令之輸入而言，且韌體將在寫入時間中處理命令且一旦被寫入至媒體便將使資料頁無效。值"01"指示資料傳送係對於當前命令之輸入而言且韌體將在不存在無效的寫入時間中處理命令。值"10"指示資料傳送係對於當前命令之輸出而言。保留值"11"。位元組51中之位元3至位元組52為媒體卡特定旗標且取決於媒體卡之類型。位元組53-63係為將來使用而保留之欄位。

媒體卡查詢命令欄位(位元組64)將用於查詢媒體卡之當前狀態及性能。已知支援包括：是否支援API協定；獲得經支援之E/D演算法；及去能API協定支援。媒體卡查詢命令傳回狀態欄位(位元組65-68)可用於保持媒體卡查詢命令之轉回值。

媒體卡特定命令長度(位元組69及70)指定命令欄位所儲存之命令位元組的總數目，其中媒體卡將執行之實際命令在媒體卡特定命令(位元組71-511)中。

附加至(多個)指令頁的將為任何資料輸入/輸出頁。調用者環境及韌體將基於所建置之密碼系統演算法而加密/解密內容。在某些狀況下，不需要將資料頁寫入至媒體，在該狀況下，韌體必須在寫入之前處理命令且接著僅將指令頁寫入至媒體，使得使用者可讀回命令並看到操作之狀態。

圖18B至圖18F給出特定媒體查詢命令之若干實例，諸如用於檢查在發送實際命令之前媒體卡是否支援API協定的API協定支援之查詢。參看圖18B及圖18C而說明特定程序。當檔案系統用於存取媒體時，將寫入一具有僅一指令

頁之內容的檔案。圖 18B 及圖 18C 之實例將說明如何經由檔案系統而查詢協定。

調用環境開啟一檔案(例如 "myAPI.bin")。為了查詢協定之支援，調用者可將 512 個位元組(圖 18B 中所示)寫入至檔案並發出一寫入檔案命令。在發出該寫入檔案命令之後，不管在將檔案寫入至媒體期間或在讀取該檔案期間，媒體皆將採取適當動作。對於該實例而言，假定在寫入之前，媒體卡處理指令頁。在此狀況下，將採取以下動作：

- a) 驗證簽名位元組
- b) 若匹配，則驗證 API 簽名
- c) 若匹配，則仔細檢查(go through)特定媒體旗標(若存在)。
- d) 檢查媒體卡查詢命令請求，其為 1
- e) 當不存在資料傳送時，驗證資料欄位及方向欄位為 0。
- f) 處理命令、更新欄位並將檔案寫入至媒體。當使用者自媒體讀取檔案(myAPI.file)時，調用者需要檢查兩個欄位：
 - (1) 韌體操作狀態：若成功完成命令操作，則此欄位將以 API_OP_SUCCESS 來更新或在失敗時以某些錯誤值來更新，且
 - (2) 僅當成功完成命令操作時才檢查媒體卡查詢命令傳回狀態。若支援協定則以 1 更新該欄位，或若不支援協定，則以 0 進行更新。見圖 18C。

若不支援協定，或卡對於此協定支援已被去能，則此等兩個可更新欄位將保持其初始預設值 0xFFFFFFFF。

特定媒體查詢命令之第二實例為一用以判定支援哪些加

密/解密演算法之查詢，其中若支援API協定且在此查詢之前該API協定為有效的，則必須檢查媒體卡。使用圖18D及18E對此進行說明。類似於API協定之支援的查詢(參看圖18B及18C而論述)，當使用檔案型系統來存取媒體時，將寫入具有僅一指令頁之內容的檔案。調用環境將寫入一檔案(例如myAPI.bin)，以使用與上文相同之實例。

調用環境將寫入一檔案；例如myAPI.bin。為查詢有關此命令之支援，使用者將包括如圖18D中所示之512個位元組且發出一寫入檔案命令。假定媒體卡在寫入之前處理指令頁，則一例示性組之動作如下：

- a)驗證簽名位元組；
- b)若匹配，則驗證API簽名；
- c)若匹配，則仔細檢查特定媒體旗標(若存在)；
- d)檢查媒體卡查詢命令請求，其為2；

e)當不存在資料傳送時，驗證資料欄位及方向欄位為0；及

f)處理命令、更新欄位並將檔案寫入至媒體。當使用者自媒體讀取檔案myAPI.bin時，調用者將檢查兩個欄位：(1)韌體操作狀態：若成功完成命令操作，則將以API_OP_SUCCESS來更新此欄位，或若其失敗，則以某些錯誤值來更新，且(2)僅當成功完成命令操作時才將檢查媒體卡查詢命令傳回狀態。此欄位將傳回32位元之整數，其中0意味著無加密/解密演算法被支援。否則將設定每一預定加密/解密演算法之每一位元。舉例而言，假定DES在位元1中，3DES在位元3中且AES在位元32中，則轉回值應為：

MSB 10000000 00000000 00000000 00001010 LSB

圖 18E 中展示該結果。注意，在例示性實施例中，演算法之最大數目為 32。基於轉回值，可判定經支援之 E/D 演算法的類型，使得可藉由在位元組 35 中設定相同預定演算法名稱而加密指令頁(自位元組 36 開始)及資料頁。注意：位元組 35 應在任何給定時間具有僅一位元設定。若設定超過 1 位元，則其將失敗。

特定媒體查詢命令之另一實例為一用以判定媒體卡是否支援 API 協定且若支援則去能該 API 協定的查詢。類似於用以判定支援哪些加密/解密演算法之查詢，將開啟一檔案，其中該檔案之內容將具有僅一指令頁。調用環境將開啟一檔案，例如再次開啟 myAPI.bin。為查詢此命令之支援，使用者可在一檔案中包括圖 18F 中所示之 512 個位元組且發出一寫入檔案命令。

在寫入檔案之後，使用者可讀取欄位並藉由首先以下檢查而驗證是否成功完成操作：(1) 韌體操作狀態：若成功完成命令操作，則將以 0 來更新此欄位，或若其失敗，則以 1 來更新；且(2) 僅當成功完成命令操作時，才將檢查媒體卡查詢命令傳回狀態。若去能協定則其將為 1，或若失敗，則其將為 0。

圖 18G 及 18H 展示基於檔案系統之特定媒體讀取命令及寫入命令的特定實例。在將韌體命令發出至媒體卡之前，檢查該卡以察看其是否支援 API 協定。如同先前協定，若將檔案系統用於存取媒體，則將開啟一具有一或多個指令頁

及任何相關聯資料頁之內容的檔案。圖 18G 中展示特定媒體寫入命令之實例。

更具體言之，假定調用者希望發出一命令(例如，SD卡命令集中之"D5")以獲得資料頁並將其寫入至某隱藏區域。調用環境將開啟一檔案且使用者將預備如圖 18G 中所示之資料頁。位元組 00-34 如同圖 18A 至圖 18F 一般，其中此實例中之指令的位元組 35 指示無加密正被使用。位元組 36-39 再次指示韌體之操作狀態，其中斜體字(此處斜體字與前述斜體字)指示欄位為韌體可更新的，且廠商或媒體識別符(該實例將廠商識別符用作"SNDK")位於位元組 40-43 中。內嵌命令接著將專用於此媒體/廠商。位元組 44-50 指示指令頁之數目(此處僅所示之頁(NIP=1))、資料頁之數目(在實例中亦為一(NDP=1))及指定為位元組數目之資料大小(此處用作區段之 512 個位元組(DSIB=512))。指令之位元組 51-68 視特定命令之需要而提供先前描述之指令資訊。

實際內嵌命令之長度(以位元組)係以位元組 69-70 而指定，其在此狀況下為含有指示至保留區域或安全區域之寫入的例示性"D5"指令之單一位元組 71。接著以零來填充配置用於特定媒體指令的剩餘區域(位元組 72-511)從而形成完整區段。實際特定命令之資料的(此處)512 個位元組係作為下一區段。

在圖 18G 中，注意：資料欄位(DATA=1)指示此指令含有"輸入"且方向欄位(Direction=00)告知韌體處理資料並將頁寫入至媒體。由於檔案系統會將資料頁寫入至媒體(如由檔

案系統所見的1024個位元組，首先512個位元組用於內嵌命令)，因此韌體將視需要以0或FF來填充資料頁。

調用者可藉由自媒體卡讀取檔案及檢查韌體操作狀態欄位而驗證操作是否已成功。若已成功完成操作，則將以API_OP_SUCCESS來更新此欄位，或若不成功，則將以某些錯誤值或API_OP_NO_SUCCESS對其進行更新。注意：由於方向為00，所以資料頁應為0或FF。

假定在寫入之前媒體卡處理指令頁，則一例示性組之動作如下：

- a)驗證簽名位元組；
- b)若簽名匹配，則驗證API簽名；
- c)檢查指令頁(來自位元組36)是否被加密；
- d)若被加密，則解密指令頁(來自位元組36)及資料頁；
- e)驗證廠商ID簽名；
- f)若廠商ID匹配，則仔細檢查特定媒體旗標(若存在)；
- g)檢查方向及資料欄位；
- h)處理實際內嵌特定媒體命令；及
- i)更新指令頁之韌體操作狀態欄位，加密(若請求)在位元組36之後的指令頁，且將其寫入至媒體。由於方向為00，所以視需要將資料頁寫為0或FF。

可以若干方式實施特定媒體讀取命令，以下段中對某些該等方式進行更多地論述。參看圖18H描述基於檔案系統之特定媒體讀取命令的實施，其之結構類似於圖18G之寫入結構。

在正自媒體讀取特定媒體命令時，該特定媒體命令現為D6且方向為10。另外，可靈活地基於使用何命令及建置之協定而修改內容。舉例而言，可告知卡寫入此檔案且該檔案具有用以告知韌體在寫入之前加密內容的特定命令。因此，所寫入之內容在寫入之前可轉換成另一形式。上述情形亦同樣適用於讀取操作。

圖18H展示其中調用者希望發出一命令(例如，SD卡命令集中之"D6")以自隱藏區域讀取一資料區段並將其置放於資料頁之下的狀況。調用環境將開啟一檔案且使用者將預備如圖18H中所示之資料頁，其中再次將廠商識別符用作"SNDK"。

資料欄位指示此為"輸出"，使得資料頁將被更新且方向欄位告知韌體在讀取期間處理資料。韌體之另一可能性為處理命令、更新資料頁並將其寫入至媒體。因此，韌體會將1024個位元組寫入至媒體。當韌體讀取檔案時，其將偵測到此為指令頁，接著其將讀取命令並適當地更新資料欄位。調用者可藉由自媒體卡讀取檔案而驗證操作是否已成功。一旦讀取檔案，調用者便可檢查韌體操作狀態欄位。若命令導致成功完成之操作，則將以API_OP_SUCCESS來更新此欄位；若未成功，則將以諸如API_OP_NO_SUCCESS之某些錯誤值對其進行更新。亦可藉由檢查DSIB欄位來驗證資料讀取之數量從而察看是否其已經更新為512。

儘管已參考特定實施例描述了本發明之各種態樣，但將瞭解，本發明係保護於附加申請專利範圍之全部範疇內。

【圖式簡單說明】

圖1為主機-卡系統之示意性表示。

圖2為主機-卡系統之方塊圖。

圖3為其中使用讀取器或硬體配接器而使卡與另一主機通信之系統的方塊圖。

圖4為在圖3系統中傳送資料及命令之先前技術的示意性表示。

圖5展示可如何根據本發明之例示性實施例內嵌特定應用程式命令。

圖6為展示韌體層之關係的例示性結構之方塊圖。

圖7為裝置驅動程式層之示意性表示。

圖8展示回應於分割狀態命令而含有的資訊。

圖9展示自變數資料區段之例示性格式。

圖10A、10B-1及10B-2展示主機側通信流程。

圖11A至11C說明主機應用程式-裝置應用程式協定。

圖12為隱藏分割應用程式之命令的狀態機圖。

圖13A至13D展示在隱藏分割協定中之一組命令的處理流程。

圖14為圖4之簡化版本。

圖15為根據本發明之態樣用於在圖3之系統中傳送資料及命令的技術之示意性表示。

圖16提供根據特定實施例之圖15之細節。

圖17A至17L給出對於若干命令的圖16之實施例之結構的詳細說明。

圖 18A 至 18H 類似於檔案層級實施例之圖 17A 至 17L。

【主要元件符號說明】

101	卡
105	韌體
151	主機
153	槽
201、203	指令
331	讀取器
333	插座
335	韌體
351	PC
401	命令
403	USB 包裝
405	命令
510	指令
511	命令部分
513a	資料部分之第一部分
513b	資料部分之第二部分
530	實際指令
531	命令部分
533	資料部分
601	命令
1001	卡/裝置
1011	記憶體

1021	FE層
1023	裝置應用程式
1025	媒體層
1027	BE層
1051	主機
1053	主機應用程式
1055	裝置驅動程式
1057	檔案系統
1059	裝置應用程式層

五、中文發明摘要：

本發明提出用於在主機與記憶卡之間傳輸特定應用程式指令之技術。將用於特定應用程式協定之命令與一簽名一起內嵌於一用於在主機與記憶卡之間通信的傳輸協定之資料部分中。此允許對特定應用程式命令之傳輸在缺失傳輸協定中的相應命令時仍可在該協定中傳輸。可在主機側的裝置驅動程式層級(level)或檔案層級實施該方法。為了實施特定應用程式協定中之讀取命令，首先將具有一內嵌之讀取命令之第一協定中之寫入命令發送至一邏輯位址，接著將一第二讀取命令發送至相同邏輯位址。

六、英文發明摘要：

十、申請專利範圍：

1. 一種操作一系統之方法，該系統包含一記憶卡及一可與該卡交換資料及命令之主機，其中在該主機與該卡之間的該資料及命令交換在傳輸過程中之某階段使用一第一協定中之一第一命令集且其中該主機及該卡包括一使用一第二協定中之一第二命令集的應用程式，該方法包含：
將一來自該第二命令集之命令內嵌於一來自該第一協定之第一指令中，該命令不具有在該第一命令集中之一相應命令；
將該第一指令自該主機傳輸至該記憶卡；及
在該卡中提取該內嵌命令。
2. 如請求項1之方法，其中使用一第二協定中之一第二命令集的該應用程式為複數個應用程式中之一者且該第二協定為複數個相應第二協定中之各別的第二協定，且其中可同時操作該複數個應用程式中的多個應用程式。
3. 如請求項1之方法，其中由該主機在檔案系統層級實施該內嵌。
4. 如請求項3之方法，其中該命令係用於將資料自該主機傳送至該卡，該資料係置放於一對應於該第一指令之檔案的一資料部分中，該方法進一步包含判定該卡上之一足以在分段之情況下容納該檔案之邏輯位址。
5. 如請求項1之方法，其進一步包含：
判定該命令是否係用於將資料自該卡傳送至該主機；
回應於判定該命令係用於將資料自該卡傳送至該主

機，設定一用於該傳送之旗標；

自該卡傳送與該命令相關聯之資料。

6. 如請求項5之方法，其中該經傳送之資料少於與該命令相關聯之全部資料。

7. 如請求項5之方法，其進一步包含：

回應於一第二命令而自該卡重新傳送與該命令相關聯的該資料。

8. 如請求項1之方法，其中由該主機在卡驅動程式層級實施該內嵌。

9. 如請求項1之方法，其中該系統進一步包含一硬體配接器，該主機與該卡經由該硬體配接器而交換該資料及該等命令，其中該主機與該硬體配接器藉由該第一協定而通信，且其中該將該第一指令自該主機傳輸至該記憶卡包含：

將該第一指令自該主機傳輸至該硬體配接器；

在該硬體配接器中將該第一指令轉譯為該第二協定中之一相應指令；及

將該相應指令自該硬體配接器傳輸至該記憶卡。

10. 如請求項1之方法，其中來自該第一協定之該指令包含一包括一來自該第一協定之命令的命令部分及一資料部分，其中該內嵌命令係內嵌於該資料部分中。

11. 如請求項10之方法，其中該資料部分進一步包括與該內嵌命令相關聯之資料。

12. 如請求項11之方法，該方法進一步包含判定包括於該資

料部分中之與該內嵌命令相關聯的該資料是否為與該內嵌命令相關聯之全部資料。

13. 如請求項12之方法，該方法進一步包含，回應於判定包括於該資料部分中之與該內嵌命令相關聯的該資料並非與該內嵌命令相關聯的全部資料，設定一旗標以用於與該內嵌命令相關聯之額外資料之接收。

14. 如請求項10之方法，其中該在該卡中提取該內嵌命令包含：

判定該資料部分包括一簽名；及

回應於確認該指令中之該簽名而自該資料部分提取該內嵌命令。

15. 如請求項10之方法，其中該相應指令包含一包括一來自該第二協定之命令的命令部分及一資料部分，其中該內嵌命令係內嵌於該資料部分中。

16. 如請求項1之方法，其中來自該第一協定之該命令係至一預定邏輯位址。

17. 如請求項16之方法，其進一步包含，在該卡中接收該經傳輸相應指令之後並在該卡中提取該內嵌命令之前：

判定該相應命令係至該預定邏輯位址；及

回應於該判定該相應命令係至該預定邏輯位址，確認該指令中之一簽名，其中回應於確認該簽名而提取該內嵌命令。

18. 如請求項17之方法，其中該判定該相應命令係至該預定邏輯位址、該確認該指令中之一簽名及該提取該內嵌命

令係由該卡之韌體加以執行。

19. 如請求項1之方法，其進一步包含執行該內嵌命令。
20. 如請求項1之方法，其中該第二協定為一SD協定。
21. 如請求項20之方法，其中該內嵌命令為一安全資料存取。
22. 如請求項1之方法，其中該第二協定為一MMC協定。
23. 如請求項1之方法，其中該第二協定為一ATA協定。
24. 如請求項1之方法，其中該第一協定為一SCSI協定。
25. 如請求項1之方法，其中該命令係用於在該主機與該卡之間傳送資料，其中該資料未快取於該主機中。
26. 一種操作一系統之方法，該系統包含一記憶卡及一可與該卡交換資料及命令之主機，其中該主機與該卡之間的該資料及命令交換在傳輸過程中之某階段使用一第一協定中之一第一命令集，且其中該主機及該卡包括一使用一第二協定中之一第二命令集的應用程式，該方法包含：
將一來自該第二命令集之命令及一簽名內嵌於一來自該第一協定之第一指令中，其中該第一指令包括一命令部分及一資料部分，且其中該簽名及來自該第二命令集之該命令係內嵌於該資料部分中；
將該第一指令自該主機傳輸至該記憶卡；
判定該第一指令之該資料部分是否含有該簽名；及
回應於該第一指令含有該簽名，自該第一指令之該資料部分提取來自該第二命令集的該命令。
27. 如請求項26之方法，其中使用一第二協定中之一第二命令集的該應用程式為複數個應用程式中之一者且該第二

協定為複數個相應第二協定中之各別第二協定，且其中可同時操作該複數個應用程式中的多個應用程式。

28. 如請求項26之方法，其中該資料部分亦包括與來自該第二命令集之該命令相關聯的資料。

29. 如請求項28之方法，該方法進一步包含判定包括於該資料部分中之與來自該第二命令集之該命令相關聯的該資料是否為與來自該第二命令集之該命令相關聯的全部資料。

30. 如請求項29之方法，該方法進一步包含，回應於判定包括於該資料部分中之與來自該第二命令集之該命令相關聯的該資料並非與來自該第二命令集之該命令相關聯的全部資料，設定一旗標以用於與來自該第二命令集之該命令相關聯的額外資料之接收。

31. 如請求項26之方法，其進一步包含：

判定該命令是否係用於將資料自該卡傳送至該主機；

回應於判定該命令係用於將資料自該卡傳送至該主機，設定一用於該傳送之旗標；

自該卡傳送與該命令相關聯之資料。

32. 如請求項31之方法，其中該經傳送之資料少於與該命令相關聯之全部資料。

33. 如請求項31之方法，其進一步包含：

回應於一第二命令而自該卡重新傳送與該命令相關聯之該資料。

34. 如請求項26之方法，其中該第一指令為一寫入。

35. 如請求項26之方法，其中來自該第二命令集之該命令為一讀取。
36. 如請求項26之方法，其中該簽名包含於該資料部分之一預定區段中。
37. 如請求項26之方法，其中該指令係用以存取在一第一邏輯位址處之資料，且其中該判定該指令之該資料部分是否含有一簽名係回應於該第一邏輯位址對應於一預定邏輯位址。
38. 如請求項26之方法，其中由該主機在該檔案驅動程式層級實施該內嵌。
39. 如請求項38之方法，其中該命令係用於將資料自該主機傳送至該卡，該資料置放於一對應於該第一指令之檔案的一資料部分中，該方法進一步包含判定該卡上之一足以在分段之情況下容納該檔案之邏輯位址。
40. 如請求項26之方法，其中由該主機在該卡驅動程式層級實施該內嵌。
41. 如請求項26之方法，其中該系統進一步包含一硬體配接器，該主機與該卡經由該硬體配接器而交換該資料及該等命令，其中該主機及該硬體配接器藉由該第一協定而通信，且其中該將該第一指令自該主機傳輸至該記憶卡包含：

將該第一指令自該主機傳輸至該硬體配接器；

在該硬體配接器中將該第一指令轉譯為該第二協定中之一相應指令；及

將該相應指令自該硬體配接器傳輸至該記憶卡。

42. 如請求項26之方法，其中來自該第一協定之該指令包含一包括一來自該第一協定之命令的命令部分及一資料部分，其中該內嵌命令係內嵌於該資料部分中。

43. 如請求項26之方法，其中來自該第一協定之該命令係至一預定邏輯位址。

44. 如請求項26之方法，其進一步包含執行該內嵌命令。

45. 如請求項26之方法，其中該第二協定為一SD協定。

46. 如請求項45之方法，其中該內嵌命令為一安全資料存取。

47. 如請求項26之方法，其中該第二協定為一MMC協定。

48. 如請求項26之方法，其中該第二協定為一ATA協定。

49. 如請求項26之方法，其中該第一協定為一SCSI協定。

50. 如請求項26之方法，其中該命令係用於在該主機與該卡之間傳送資料，其中該資料未快取於該主機中。

51. 一種操作一非揮發性記憶卡之方法，其包含：

接收一指令，其中該指令包括一命令部分及一資料部分；

判定該指令之該資料部分是否含有一簽名；及

回應於該指令含有該簽名而自該指令之該資料部分提取一命令。

52. 如請求項51之方法，其進一步包含：

判定該命令是否係用於自該卡傳送資料；

回應於判定該命令係用於自該卡傳送資料，設定一用於該傳送之旗標；

自該卡傳送與該命令相關聯的資料。

53. 如請求項52之方法，其中該經傳送之資料少於與該命令相關聯的全部資料。

54. 如請求項53之方法，其進一步包含：

回應於一第二命令而自該卡重新傳送與該命令相關聯的該資料。

55. 如請求項51之方法，其中該資料部分進一步包括與該經提取命令相關聯之資料。

56. 如請求項55之方法，該方法進一步包含判定包括於該資料部分中之與該經提取命令相關聯的該資料是否為與該經提取命令相關聯之全部資料。

57. 如請求項56之方法，該方法進一步包含，回應於判定包括於該資料部分中之與該經提取命令相關聯的該資料並非與該經提取命令相關聯之全部資料，設定一旗標以用於與該經提取命令相關聯之額外資料之接收。

58. 如請求項51之方法，其中該簽名包含於該資料部分之一預定區段中。

59. 如請求項51之方法，其中該指令係在一具有一第一命令集之第一協定中，且其中該經提取之命令係來自一第二協定中之一第二命令集且不具有在該第一命令集中的一相應命令。

60. 如請求項59之方法，其中該第二協定用於複數個應用程式中之一者且該第二協定為複數個相應應用程式協定中之各別應用程式協定，且其中可同時操作該複數個應用

程式中的多個應用程式。

61. 如請求項51之方法，其中該指令係用以存取在一第一邏輯位址處之資料且其中該判定該指令之該資料部分是否含有一簽名係回應於該第一邏輯位址對應於一預定邏輯位址。
62. 如請求項51之方法，其中該判定該指令含有一簽名及該提取一來自該指令之該資料部分之命令係由該卡之韌體來執行。
63. 如請求項51之方法，其進一步包含執行該經提取之命令。
64. 如請求項51之方法，其中該指令在一SD協定中。
65. 如請求項51之方法，其中該經提取之命令為一安全資料存取。
66. 如請求項51之方法，其中該指令在一MMC協定中。
67. 如請求項51之方法，其中該指令在一ATA協定中。
68. 一種傳輸一來自一第一協定之命令的方法，其包含：
 形成一來自一第二協定之指令，該指令包括一接受一相關聯資料部分的命令；
 將來自該第一協定之該命令及一指示該指令含有一來自該第一協定之命令的簽名置放於該指令之該相關聯資料部分中；及
 傳輸該指令。
69. 如請求項68之方法，其中該第一協定係用於複數個應用程式中之一者且該第一協定為複數個相應應用程式協定中之各別應用程式協定，且其中可同時操作該複數個應

用程式中的多個應用程式。

70. 如請求項68之方法，其進一步包含將與來自該第一協定之該命令相關聯之資料置放於該相關聯的資料部分中。
71. 如請求項70之方法，該方法進一步包含判定包括於該資料部分中之與來自該第一協定之該命令相關聯的該資料是否為與來自該第一協定之該命令相關聯的全部資料。
72. 如請求項71之方法，該方法進一步包含：回應於判定包括於該資料部分中之與來自該第一協定之該命令相關聯的該資料並非與來自該第一協定之該命令相關聯之全部資料，設定一旗標以用於與來自該第一協定之命令相關聯的額外資料之接收。
73. 如請求項68之方法，其中來自該第二協定之該指令為一寫入。
74. 如請求項68之方法，其中來自該第一協定之該命令為一讀取。
75. 如請求項68之方法，其中將該簽名置放於該相關聯資料部分之一預定區段中。
76. 如請求項68之方法，其中由該主機在該檔案驅動程式層級實施該內嵌。
77. 如請求項68之方法，其中該相關聯資料部分係形成為一檔案之部分。
78. 如請求項68之方法，其中將來自該第二協定之該指令傳輸至一預定邏輯位址。
79. 如請求項68之方法，其中來自該第一協定之該命令係用

於該資料傳送且其中該資料未快取於該傳輸裝置中。

80. 一種操作一系統之方法，該系統包含一記憶卡及一可與該卡交換資料及命令之主機，其中在該主機與該卡之間的該資料及命令交換在傳輸過程中之某階段使用一第一協定中之一第一命令集且，其中該主機及該卡包括一使用一第二協定中之一第二命令集的應用程式，該方法包含：

將一來自該第二命令集之命令內嵌於一來自該第一協定之第一指令中；

將該第一指令自該主機傳輸至該記憶卡中之一第一邏輯位址；

判定該內嵌命令是否係用於將資料自該卡傳送至該主機；

隨後將一第二指令自該主機傳輸至該記憶卡；

回應於判定該內嵌命令係用於將資料自該卡傳送至該主機，判定該第二指令是否係至該記憶卡中之該第一邏輯位址；及

回應於該第二指令係至該記憶卡中之該第一邏輯位址，將資料自該卡傳送至該主機。

81. 如請求項80之方法，其中該第一指令為一寫入。

82. 如請求項80之方法，其中來自該第二指令之該命令為一讀取。

83. 如請求項80之方法，其中該簽名包含於該第一指令之一預定區段中。

84. 如請求項80之方法，其中由該主機在該檔案驅動程式層級實施該內嵌。

85. 如請求項80之方法，其中該命令係用於在該主機與該卡之間傳送資料，其中該資料未快取於該主機中。

86. 一種記憶卡結構，其包含：

一非揮發性記憶體；

一前端層，其用於根據一第一協定與一與該卡相連接的主機交換指令，其中第一層判定該第一協定中之一傳入指令的資料部分是否含有一簽名，且用於回應於判定無該簽名而根據該第一協定中的該指令來存取該記憶體；及

一應用程式層，其中回應於判定該第一協定中之一傳入指令含有一簽名，將該傳入指令自該前端層傳送至該應用程式，其中自該傳入指令之該資料部分提取一第二協定中之一指令且根據該第二協定中的該指令而存取該記憶體。

87. 如請求項86之記憶卡結構，其中該簽名係包含於該資料部分之一預定區段中。

88. 如請求項86之記憶卡結構，其中該指令係在一具有一第一命令集之第一協定中，且其中該經提取之命令係來自一第二協定中之一第二命令集且不具有該第一命令集中之一相應命令。

89. 如請求項86之記憶卡結構，其中該指令係用以存取在一第一邏輯位址處之資料且其中該判定該指令之該資料部分是否含有一簽名係回應於該第一邏輯位址對應於一預

定邏輯位址。

90. 如請求項86之記憶卡結構，其中該前端層及該應用程式層係以韌體予以實施。
91. 如請求項86之記憶卡結構，其中該記憶卡結構可同時處理多個此等指令。
92. 如請求項91之記憶卡結構，其中該應用程式層包括多個應用程式且其中經同時處理之該等多個指令來自該等應用程式中的一個以上應用程式。
93. 如請求項86之記憶卡結構，其中回應於判定該傳入指令係用於自該卡之資料傳送，該應用程式層設定一用於該傳送之旗標並自該卡傳送與該傳入指令相關聯的資料。
94. 如請求項93之記憶卡結構，其中回應於一第二傳入指令而自該卡重新傳送與該傳入指令相關聯之該資料。
95. 如請求項86之記憶卡結構，其中該應用程式層判定該資料部分是否進一步包括與該第二協定中之該指令相關聯的資料。
96. 如請求項95之記憶卡結構，其中該應用程式層判定包括於該資料部分中之與該第二協定中之該指令相關聯的該資料是否為與該第二協定中之該指令相關聯之全部資料。
97. 如請求項96之記憶卡結構，其中回應於判定包括於該資料部分中之與該第二協定中之該指令相關聯的該資料並非與該第二協定中之該指令相關聯之全部資料，該應用程式層設定一旗標以用於與該第二協定中之該指令相關聯的額外資料之接收。

十一、圖式：

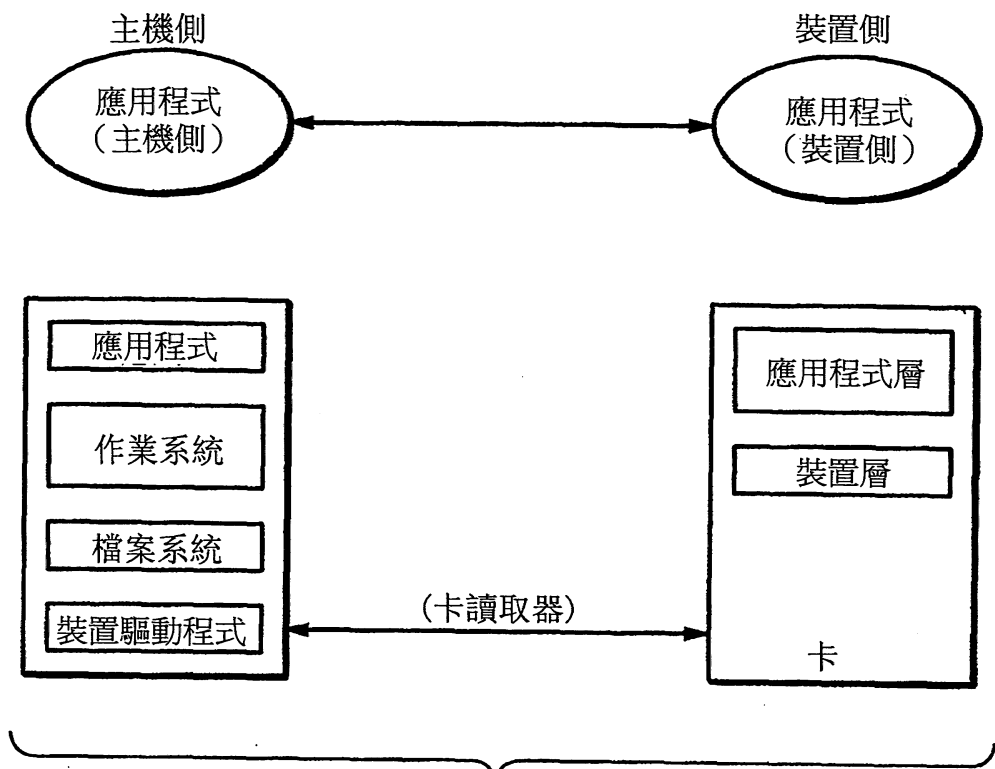


圖1

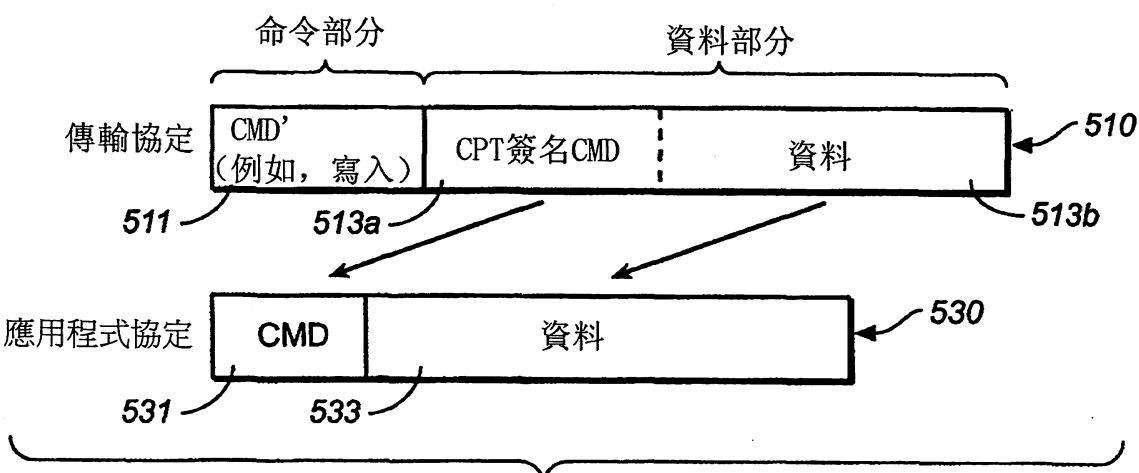


圖5

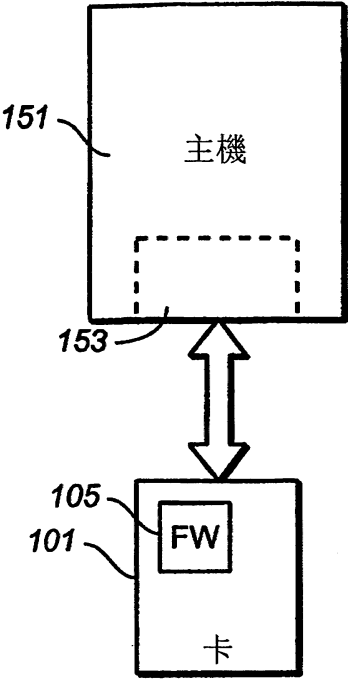


圖2

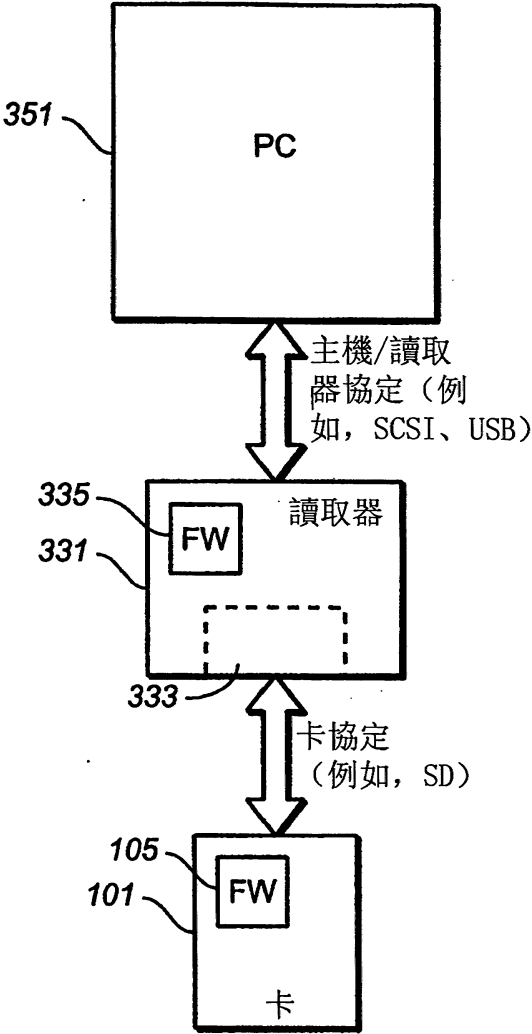


圖3

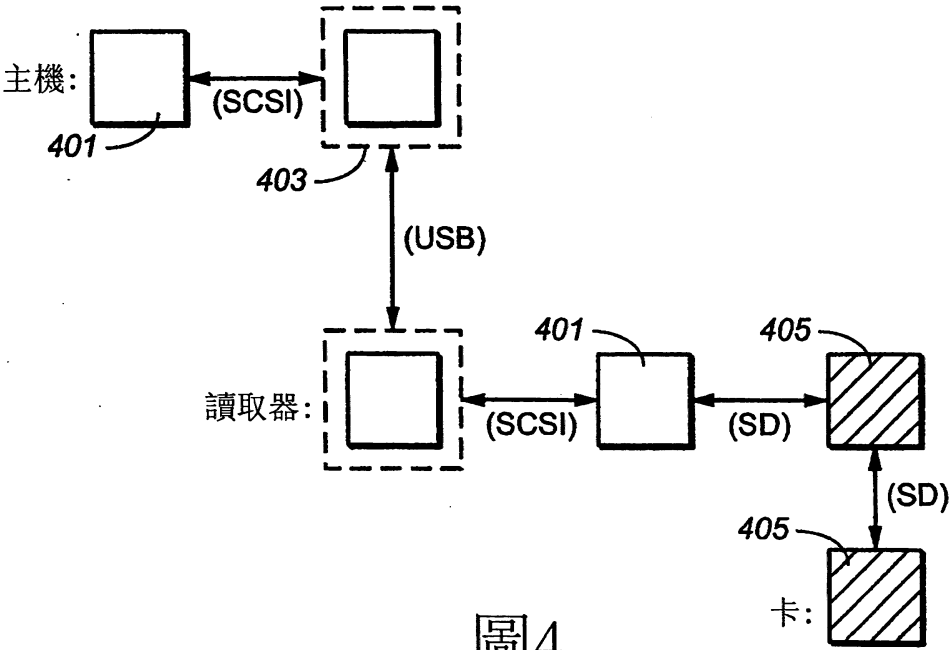


圖4

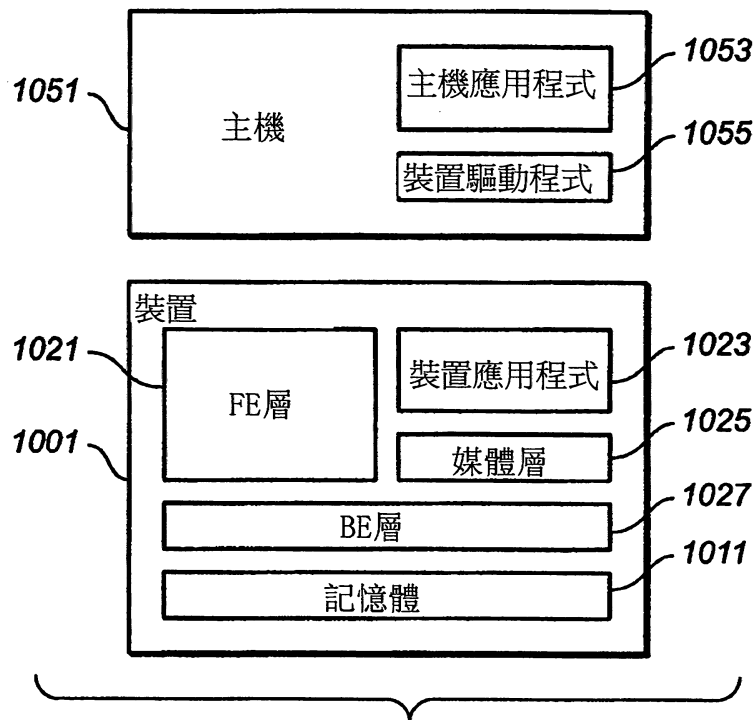


圖6

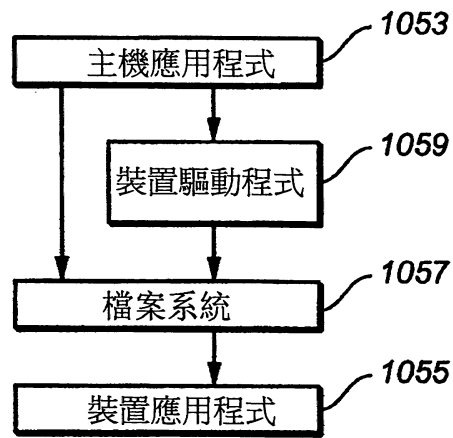


圖7

位元 組索引	描述	實例
0	裝置應用程式 分割 ID	0x01
1-4	以區段之分割大小	0x00005000區段-10MB
5	分割： • 關閉=0 • 開啟=1	0x00-關閉
6-9	裝置應用程式版本	0x00000000
10	最末裝置操作狀 態： • 正確=0x00 • 錯誤≠0x00	0x00
11-511	保留	NA

圖8

位元 組索引	描述	
0-31	應用程式傳遞簽名	"支援傳遞模式"
32	應用程式 ID	
33	應用程式命令操作碼 索引	
34-511	應用程式命令自變數 資料	

圖9

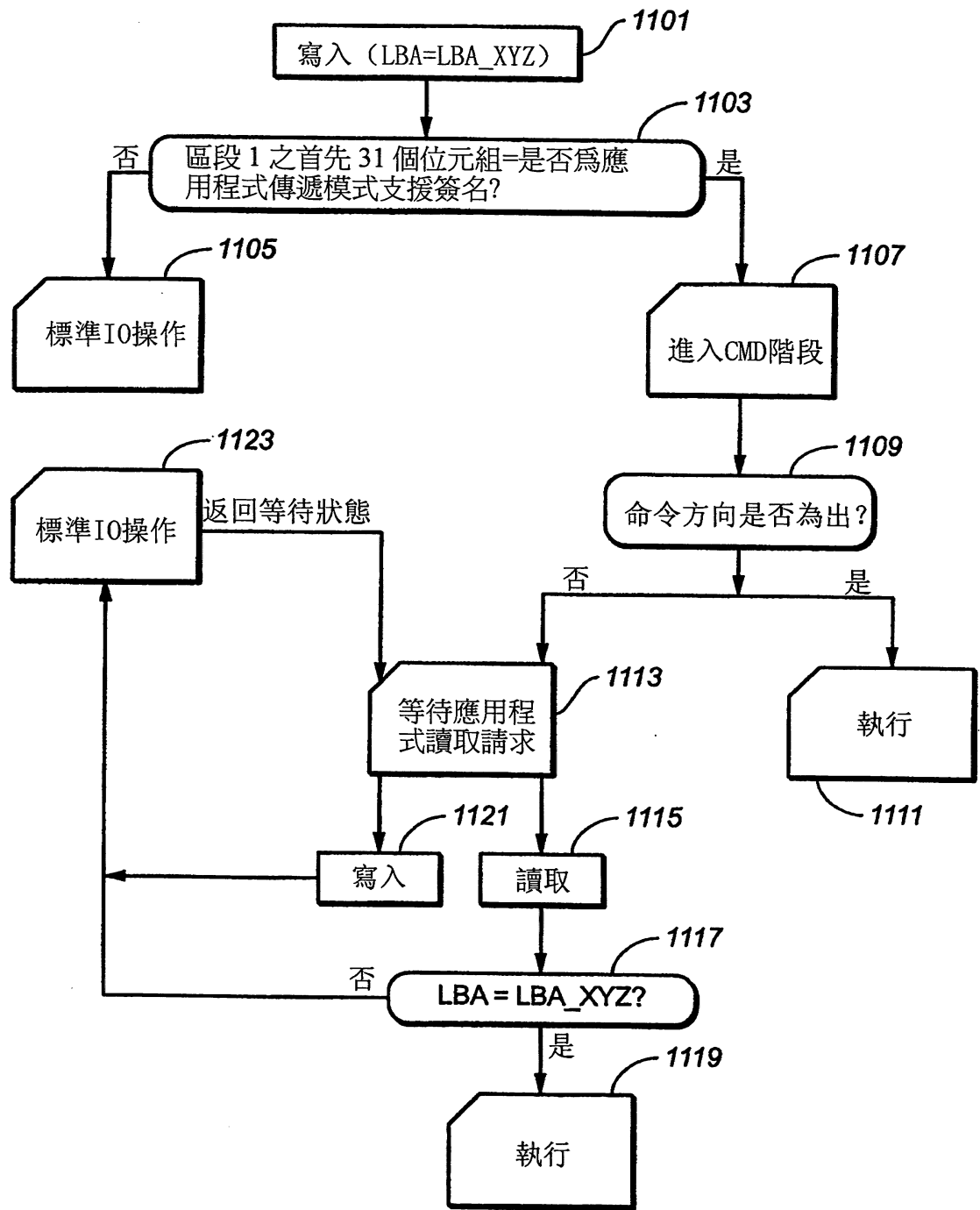


圖10A

圖10B

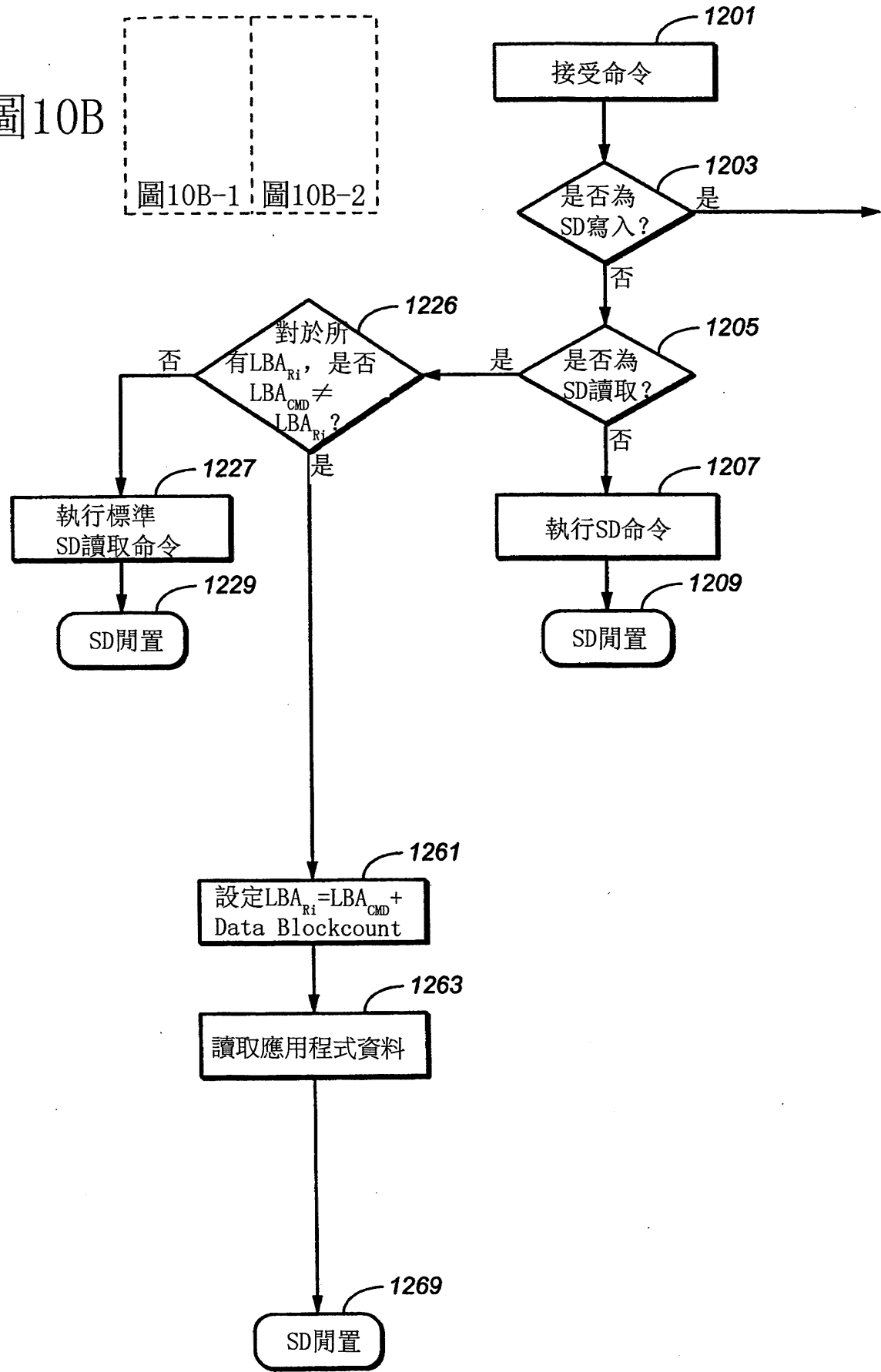
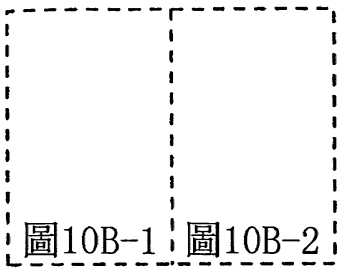


圖10B-1

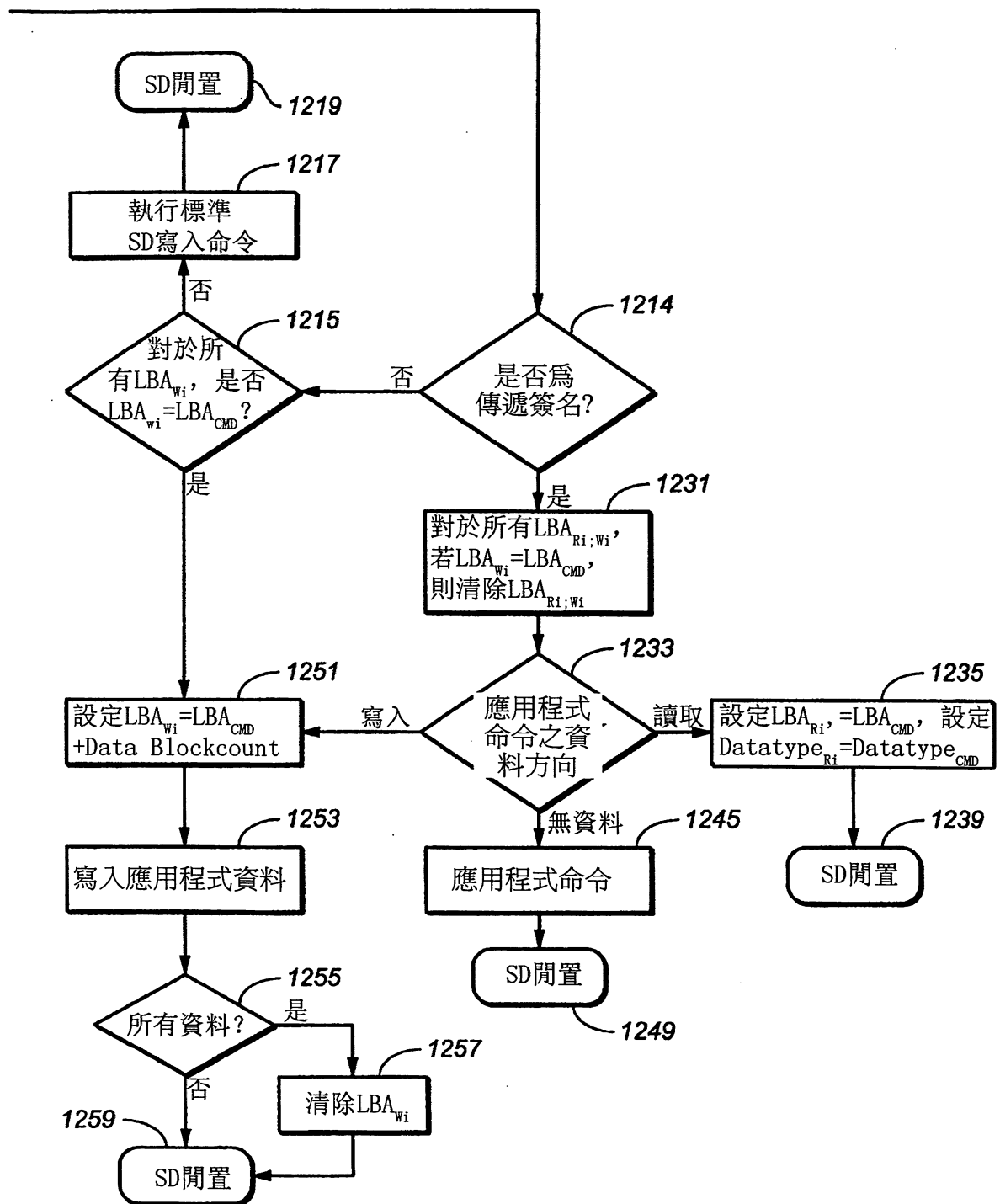


圖10B-2

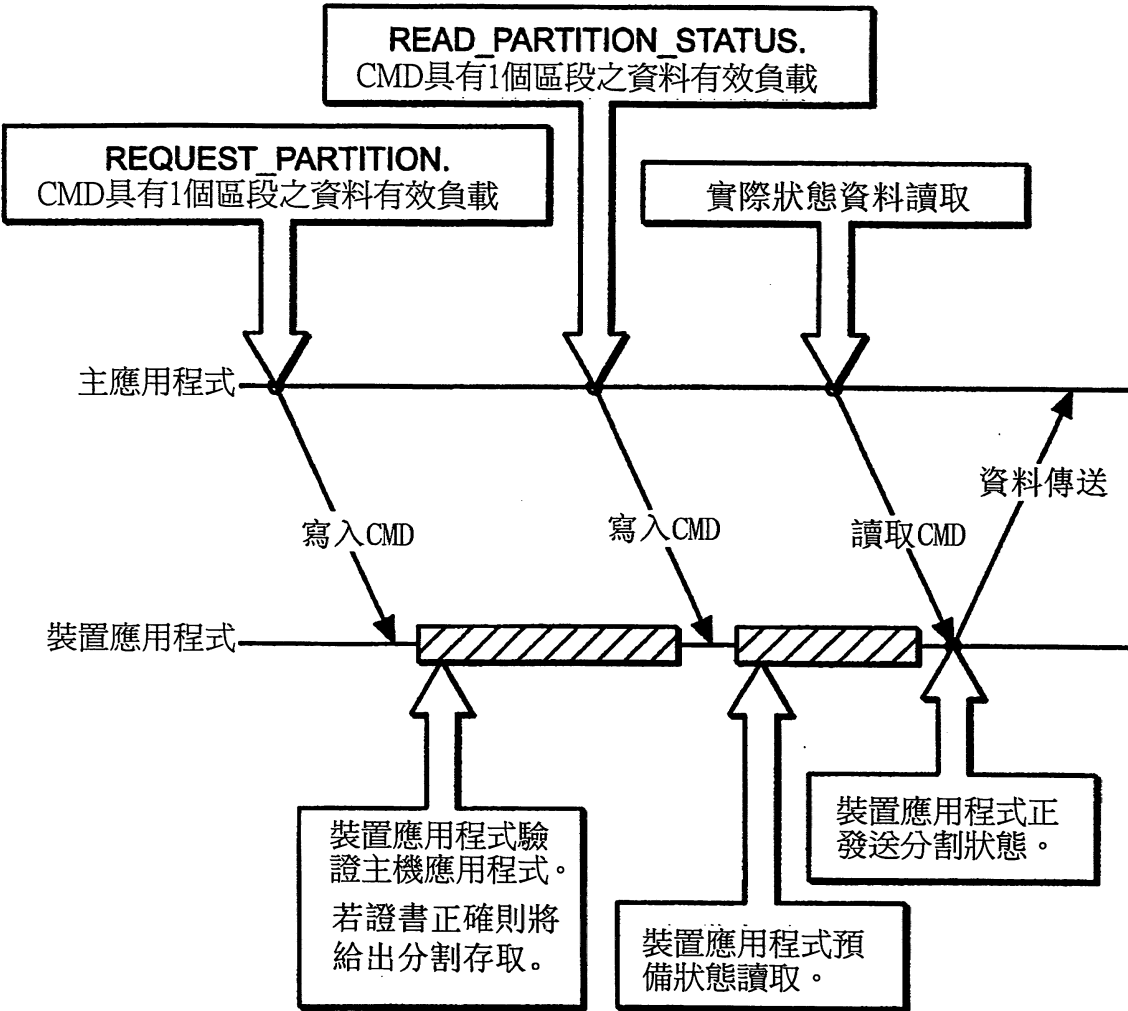


圖11A

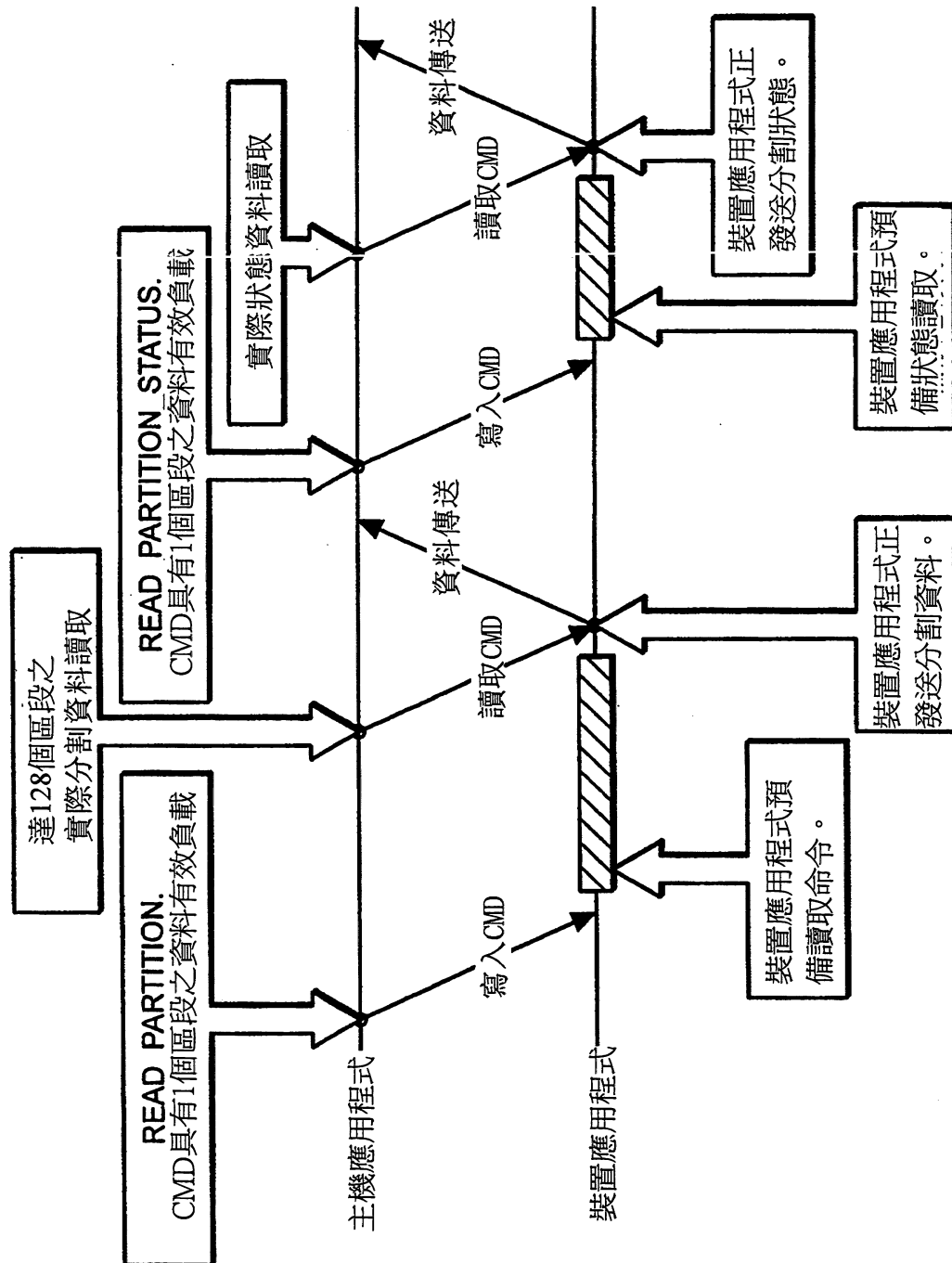


圖 11B

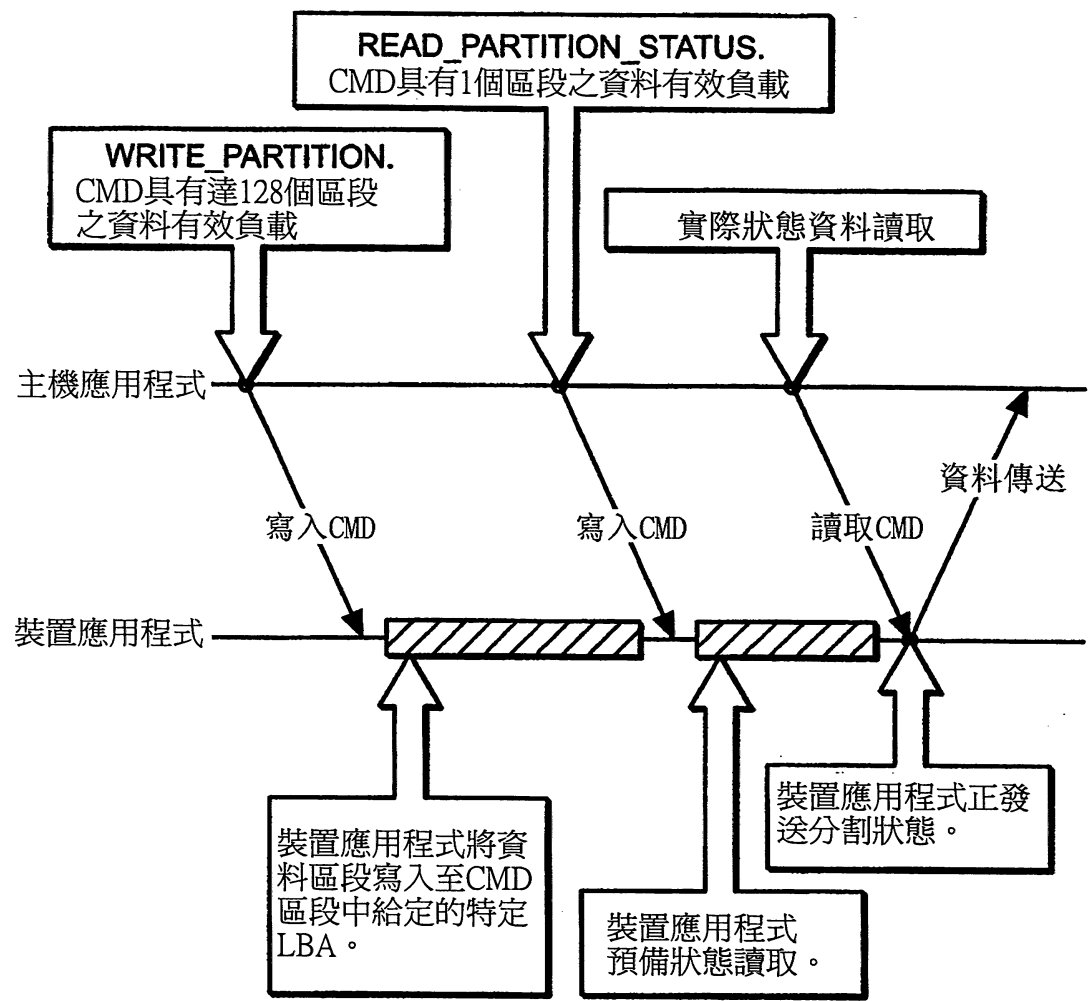


圖11C

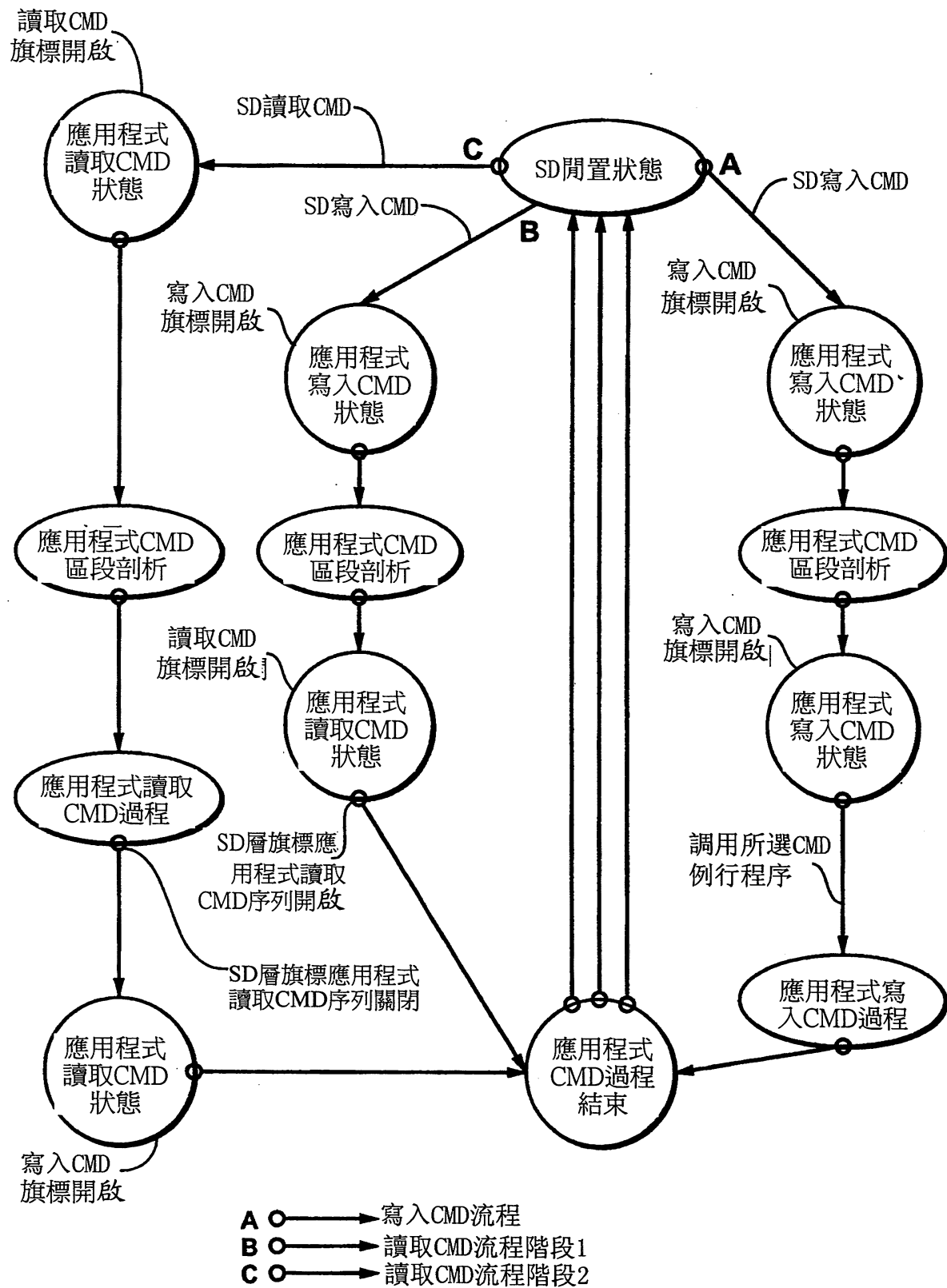
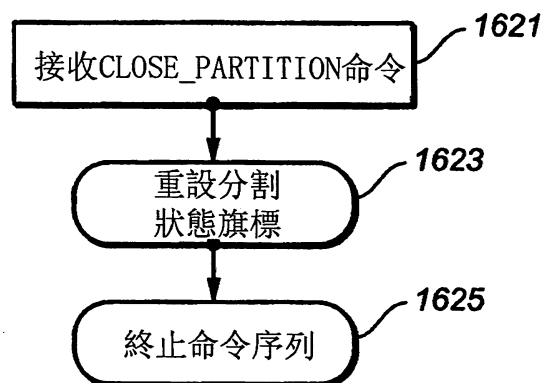
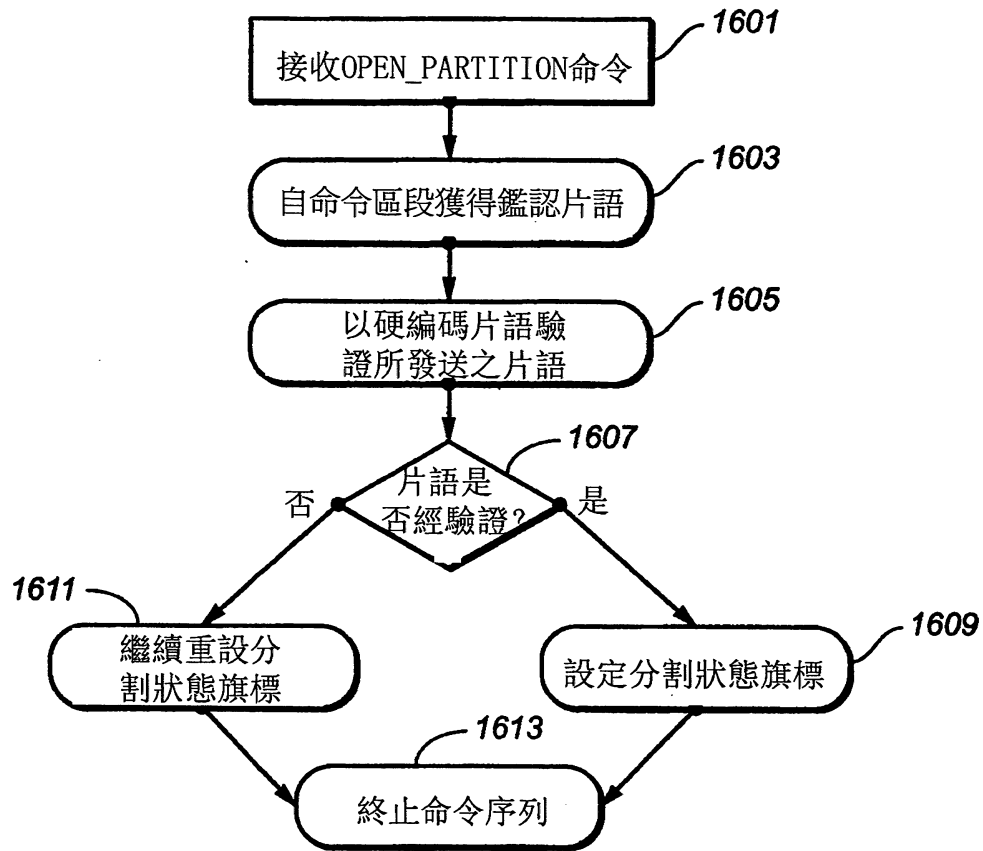


圖12



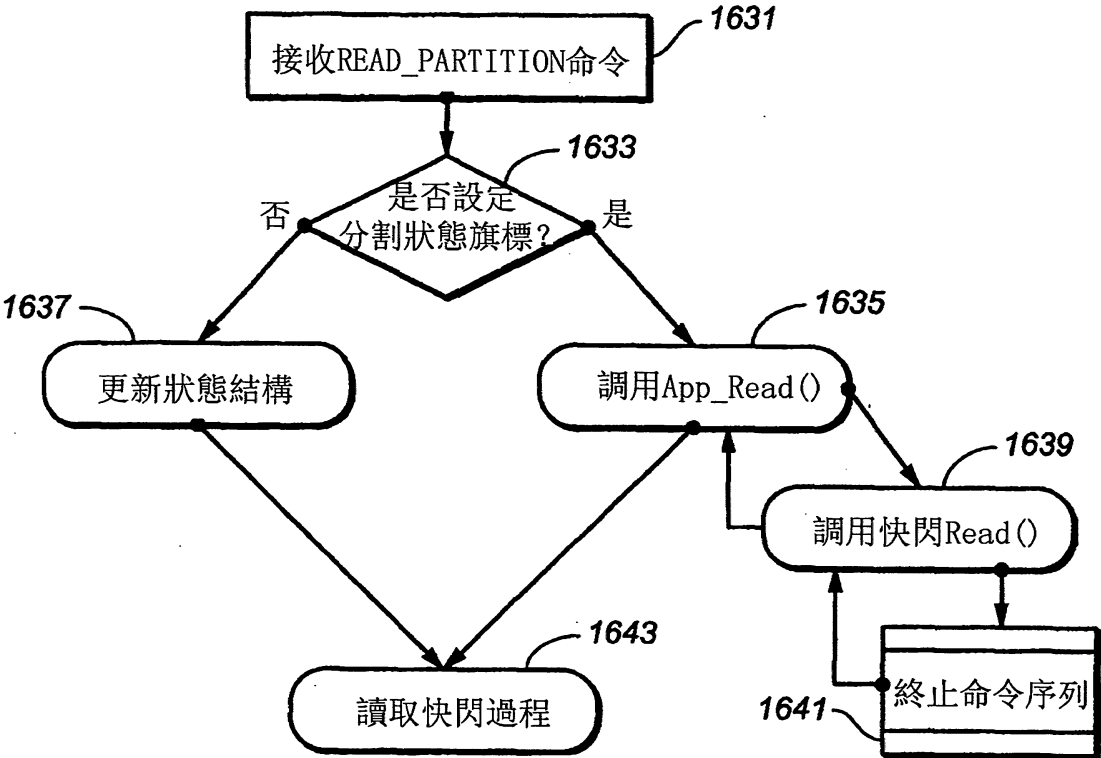


圖13C

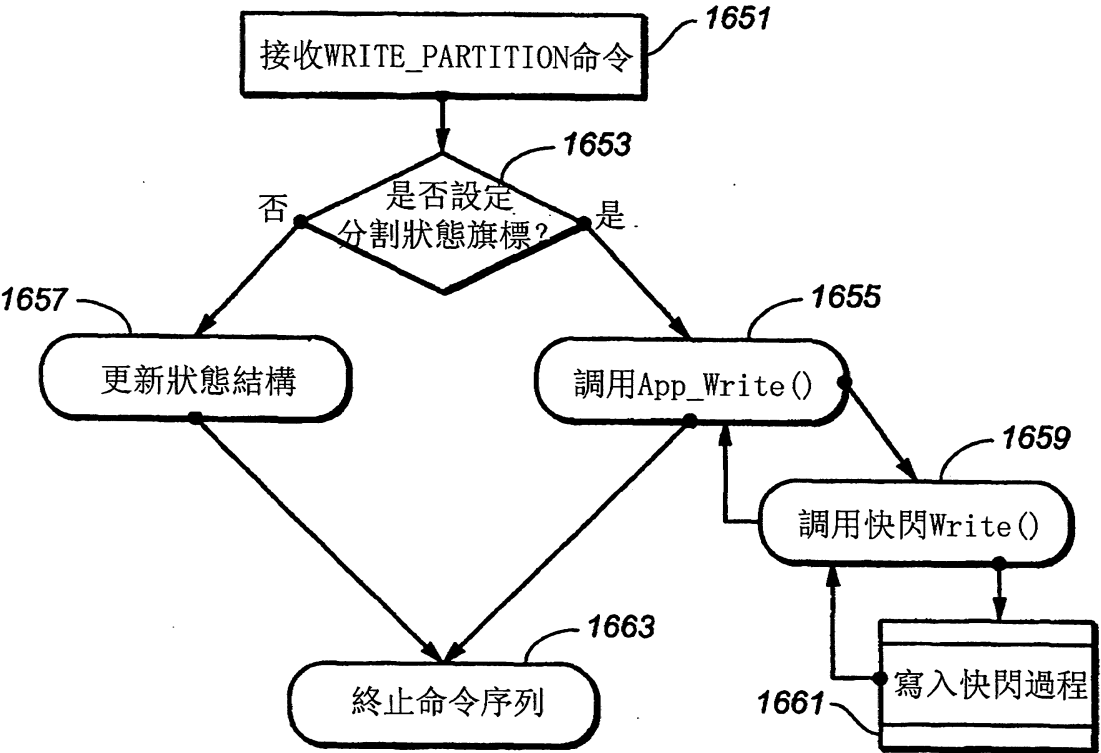
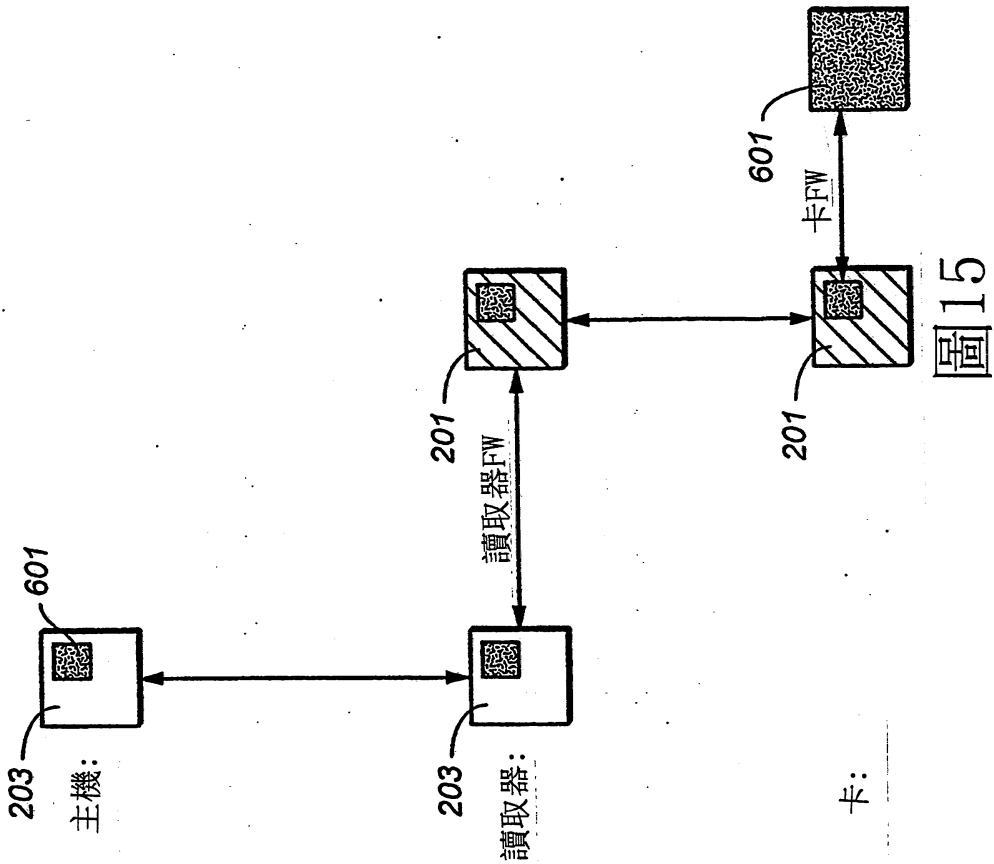
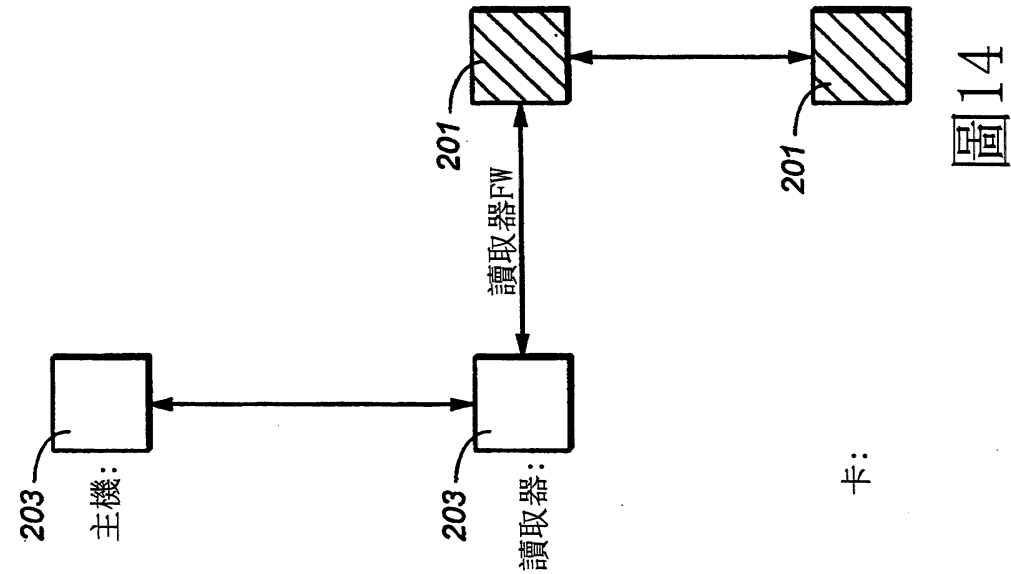


圖13D



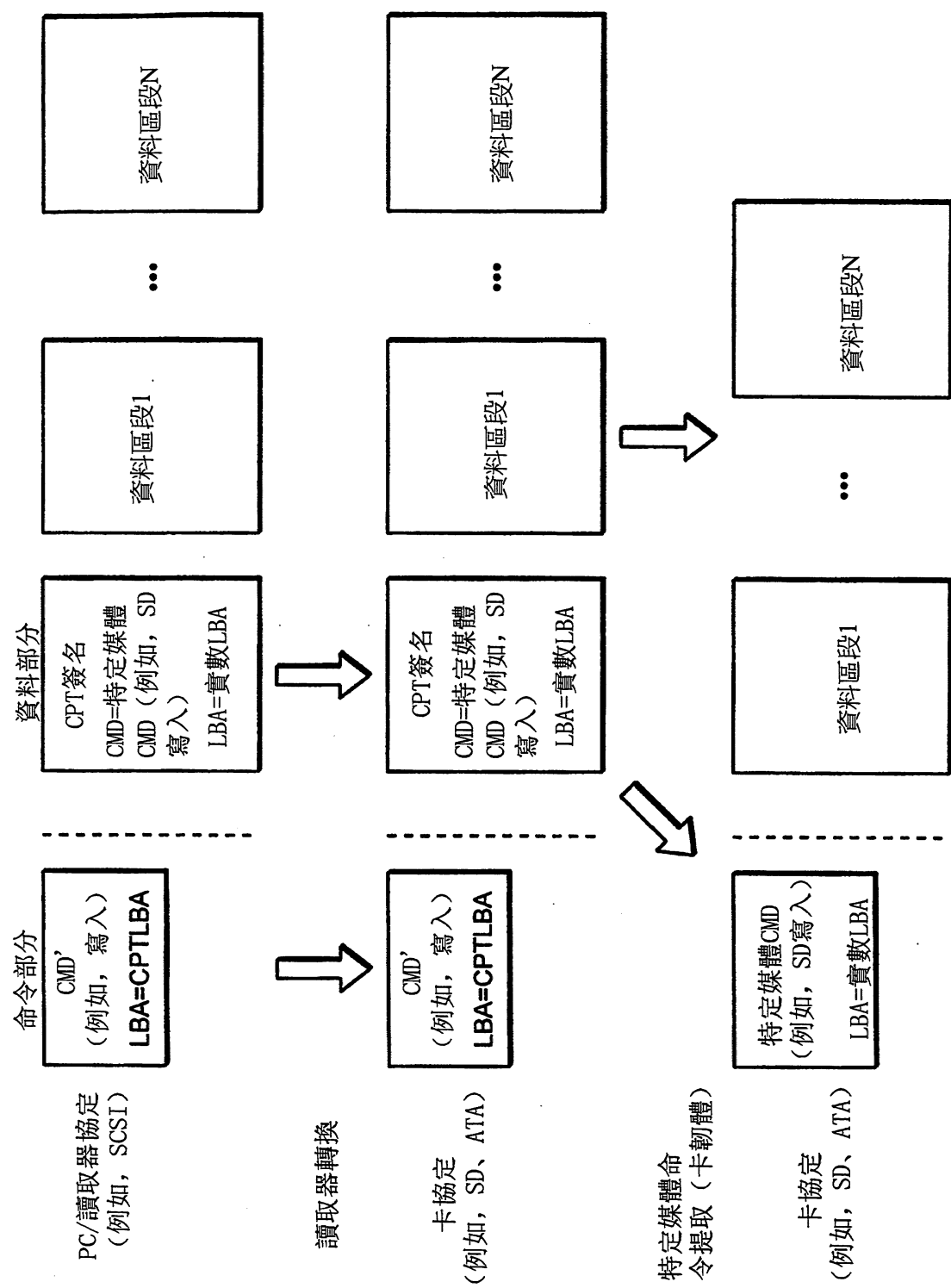


圖16

位元	7	6	5	4	3	2	1	0
位元組								
0-31	CPT簽名							
32	命令							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	媒體卡特定旗標						資料	Dir
37	媒體卡特定旗標							
38-47	保留							
48-511	媒體卡特定命令							

圖17A

位元	7	6	5	4	3	2	1	0
位元組								
0-31	"支援卡傳遞模式"							
32-511	0							

圖17B

位元	7	6	5	4	3	2	1	0
位元組								
0-31	"卡傳遞模式檢查"							
32	命令							
33	0							
34	0							
35	0							
36	0							
37	0						1	0
38-47	0							
48-511	0							

圖17C

位元	7	6	5	4	3	2	1	0
位元組								
0-31	"支援卡傳遞模式"							
32-511	0							

圖17D

位元	7	6	5	4	3	2	1	0
位元組								
0-31	卡傳遞輸出命令							
32	1							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	媒體卡特定旗標						1	1
37	媒體卡特定旗標							
38-47	保留							
48-511	媒體卡特定命令							

圖17E

資料位元組0
資料位元組1
資料位元組2
資料位元組3
...
...
...
資料位元組"資料傳送長度"-1

圖17F

位元	7	6	5	4	3	2	1	0
位元組								
0-31	卡傳遞輸入命令							
32	2							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	媒體卡特定旗標						1	0
37	媒體卡特定旗標							
38-47	保留							
48-511	媒體卡特定命令							

圖17G

位元	7	6	5	4	3	2	1	0
位元組								
0-31	卡傳遞無I/O命令							
32	3							
33	0							
34	0							
35	0							
36	媒體卡特定旗標						0	0
37	媒體卡特定旗標							
38-47	保留							
48-511	媒體卡特定命令							

圖17H

位元	7	6	5	4	3	2	1	0
位元組								
0-31	卡傳遞狀態讀取							
32	6							
33	0							
34	0							
35	0							
36	媒體卡特定旗標						1	0
37	媒體卡特定旗標							
38-47	保留							
48-511	0							

圖17I

位元	7	6	5	4	3	2	1	0
位元組								
0-31	CPT簽名							
32	命令							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	保留		BLKH APP		回應類型			Dir
37-47	保留							
48	開始位元	傳輸位元	命令索引					
49-52	自變數							
53	CRC7							終止位元
54-511	0							

圖17J

位元	7	6	5	4	3	2	1	0
位元組								
0-31	卡傳遞狀態讀取							
32	6							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	保留						1	0
37-511	0							

圖17K

卡傳遞狀態讀取

位元	7	6	5	4	3	2	1	0
位元組								
0-31	CPT簽名							
32	命令							
33	資料傳送長度 (MSB)							
34	資料傳送長度 (LMSB)							
35	資料傳送長度 (LSB)							
36	保留						資料	Dir
37-47	保留							
48	錯誤暫存器							
49	區段計數暫存器/特徵暫存器							
50	區段暫存器							
51	磁柱低暫存器							
52	磁柱高暫存器							
53	裝置/磁頭暫存器							
54	命令/狀態暫存器							
55-511	0							

圖17L

位元	7	6	5	4	3	2	1	0
位元組								
00-01	簽名位元組							
02-34	API簽名							
35	加密/解密資訊							
36-39	韌體操作狀態							
40-43	廠商識別符							
44	指令頁之數目 (NIP)							
45-46	資料頁之數目 (NDP)							
47-50	以位元組之資料大小							
51	媒體卡特定旗標					方向		資料
52	媒體卡特定旗標							
53-63	保留							
64	媒體卡1-是否支援協定; 2-獲得支援E/D演算法				查詢命令3-去 能寫入協定支援			
65-68	媒體卡查詢命令傳回狀態							
69-70	以位元組之媒體卡特定命令長度							
71-511	媒體卡特定命令							

圖 18A

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	加密/解密資訊=0							
36-39	韌體操作狀態=0xFFFFFFFF							
40-43	廠商識別符=0							
44	NIP = 1							
45-46	NDP = 0x0000							
47-50	DSIB = 0x00000000							
51	媒體卡特定旗標						0	0
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=1（是否支援協定）							
65-68	媒體卡查詢命令傳回狀態=0xFFFFFFFF							
69-511	0							

圖 18B

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	加密/解密資訊=0							
36-39	韌體操作狀態= API OP SUCCESS或錯誤值							
40-43	廠商識別符=0							
44	NIP = 1							
45-46	NDP = 0x0000							
47-50	DSIB = 0x00000000							
51	媒體卡特定旗標					0		0
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=1（是否支援協定）							
65-68	媒體卡查詢命令傳回狀態=1或0							
69-511	0							

圖18C

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	加密/解密資訊=0							
36-39	韌體操作狀態=0xFFFFFFFF							
40-43	廠商識別符=0							
44	NIP = 1							
45-46	NDP = 0x0000							
47-50	DSIB = 0x00000000							
51	媒體卡特定旗標					0		0
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=2（獲得支援E/D演算法）							
65-68	媒體卡查詢命令傳回狀態=0xFFFFFFFF							
69-511	0							

圖18D

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	加密/解密資訊=0							
36-39	韌體操作狀態= API OP SUCCESS或錯誤值							
40-43	廠商識別符=0							
44	NIP = 1							
45-46	NDP = 0x0000							
47-50	DSIB = 0x00000000							
51	媒體卡特定旗標					0		0
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=2（獲得支援E/D演算法）							
65-68	媒體卡查詢命令傳回狀態=0或不帶正負號的整數值							
69-511	0							

圖18E

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	加密/解密資訊=0							
36-39	韌體操作狀態=0xFFFFFFFF							
40-43	廠商識別符=0							
44	NIP = 1							
45-46	NDP = 0x0000							
47-50	DSIB = 0x00000000							
51	媒體卡特定旗標					0		0
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=3（去能API協定支援）							
65-68	媒體卡查詢命令傳回狀態=0xFFFFFFFF							
69-511	0							

圖18F

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	API 無加密							
36-39	韌體操作狀態=0xFFFFFFFF							
40-43	廠商識別符= “SNDK”							
44	NIP = 1							
45-46	NDP = 1							
47-50	DSIB = 512							
51	媒體卡特定旗標=0				方向=00		資料=1	
52	媒體卡特定旗標							
53-63	保留=0							
64	媒體卡查詢命令=API NO INQUIRY							
65-68	媒體卡查詢命令傳回狀態=0							
69-70	以位元組之媒體卡查詢命令長度=1							
71	媒體卡查詢命令= D5							
72-511	0							
資料位元組0								
資料位元組1								
資料位元組2								
資料位元組3								
...								
...								
...								
資料位元組512								

圖18G

位元	7	6	5	4	3	2	1	0
位元組								
00-01	1973							
02-34	"進階程式介面"							
35	API無加密							
36-39	韌體操作狀態=0xFFFFFFFF							
40-43	廠商識別符= “SNDK”							
44	NIP = 1							
45-46	NDP = 1							
47-50	DSIB = 0xFFFFFFFF							
51	媒體卡特定旗標=0					方向=10		資料=1
52	媒體卡特定旗標=0							
53-63	保留=0							
64	媒體卡查詢命令=API NO INQUIRY							
65-68	媒體卡查詢命令傳回狀態=0							
69-70	以位元組之媒體卡查詢命令長度=1							
71	媒體卡查詢命令= D6							
72-511	0							
資料位元組0								
資料位元組1								
資料位元組2								
資料位元組3								
...								
...								
...								
資料位元組512								

圖18H

七、指定代表圖：

(一)本案指定代表圖為：第(5)圖。

(二)本代表圖之元件符號簡單說明：

510	指令
511	命令部分
513a	資料部分之第一部分
513b	資料部分之第二部分
530	實際指令
531	命令部分
533	資料部分

八、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

(無)

三、發明人：(共 6 人)

姓 名：(中文/英文)

1. 羅伯特 C 張
CHANG, ROBERT C.
2. 亨利 利卡多 哈頓
HUTTON, HENRY RICARDO
3. 法西德 沙貝特-沙爾西
SABET-SHARGHI, FARSHID
4. 哈路克 肯特 泰尼克
TANIK, HALUK KENT
5. 羅 巴利列
BARZILAI, RON
6. 美托 丹 阿里
ARI, MEYTAL DAM

國 籍：(中文/英文)

1. 美國 U.S.A.
2. 美國 U.S.A.
3. 美國 U.S.A.
4. 土耳其 TURKEY
- 5 以色列 ISRAEL
6. 以色列 ISRAEL