

(21)申請案號：102116202

(22)申請日：中華民國 102 (2013) 年 05 月 07 日

(51)Int. Cl. : G06F9/44 (2006.01) G06F9/48 (2006.01)

(30)優先權：2012/06/01 日本 2012-125920

(71)申請人：新力股份有限公司 (日本) SONY CORPORATION (JP)

日本

(72)發明人：中村祐一 NAKAMURA, YUUICHI (JP)；新田尚史 NITTA, TAKASHI (JP)；永尾制一 NAGAO, SADAKAZU (JP)；岡山貴広 OAKAYAMA, TAKAHIRO (JP)；中田大輔 NAKATA, DAISUKE (JP)；上原朋樹 UEHARA, TOMOKI (JP)

(74)代理人：林志剛

申請實體審查：有 申請專利範圍項數：21 項 圖式數：19 共 72 頁

(54)名稱

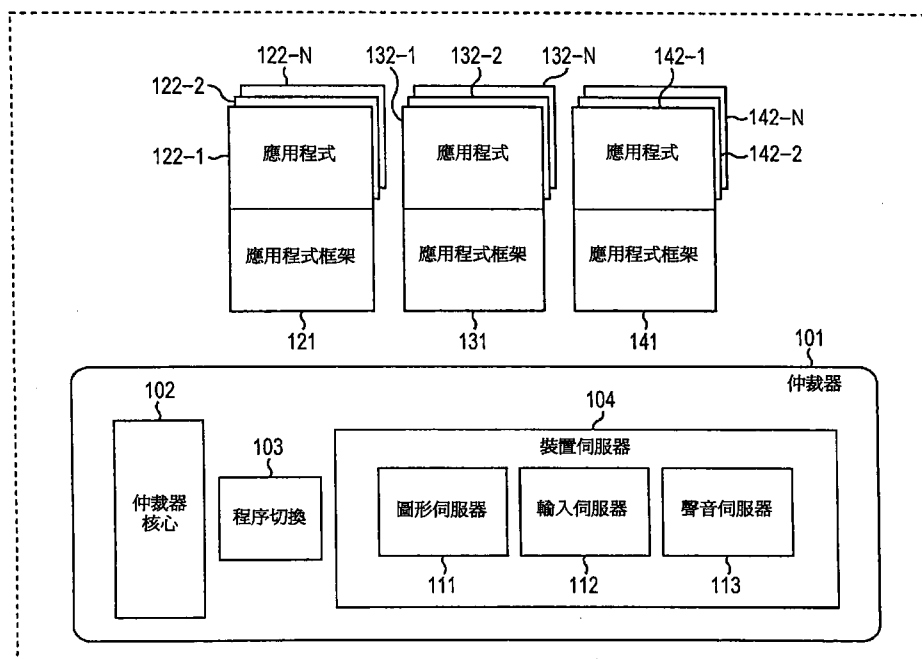
資訊處理裝置、資訊處理方法及程式

INFORMATION PROCESSING APPARATUS, INFORMATION PROCESSING METHOD, AND PROGRAM

(57)摘要

提出一種資訊處理裝置，包括複數個應用程式框架，在複數個應用程式框架上執行應用程式，以及一決定單元，配置以控制複數個應用程式框架之可操作狀態的切換。

第 3 圖



101：仲裁器

102：仲裁器核心

103：程序切換

104：裝置伺服器

111：圖形伺服器

112：輸入伺服器

113：聲音伺服器

121：應用程式框架

122-1-122-N：應用程式

131：應用程式框架

132-1-132-N：應用程式

141：應用程式框架

142-1-142-N：應用程式

(21)申請案號：102116202

(22)申請日：中華民國 102 (2013) 年 05 月 07 日

(51)Int. Cl.：

G06F9/44 (2006.01)

G06F9/48 (2006.01)

(30)優先權：2012/06/01

日本

2012-125920

(71)申請人：新力股份有限公司 (日本) SONY CORPORATION (JP)

日本

(72)發明人：中村祐一 NAKAMURA, YUUICHI (JP)；新田尚史 NITTA, TAKASHI (JP)；永尾制一 NAGAO, SADAKAZU (JP)；岡山貴広 OAKAYAMA, TAKAHIRO (JP)；中田大輔 NAKATA, DAISUKE (JP)；上原朋樹 UEHARA, TOMOKI (JP)

(74)代理人：林志剛

申請實體審查：有 申請專利範圍項數：21 項 圖式數：19 共 72 頁

(54)名稱

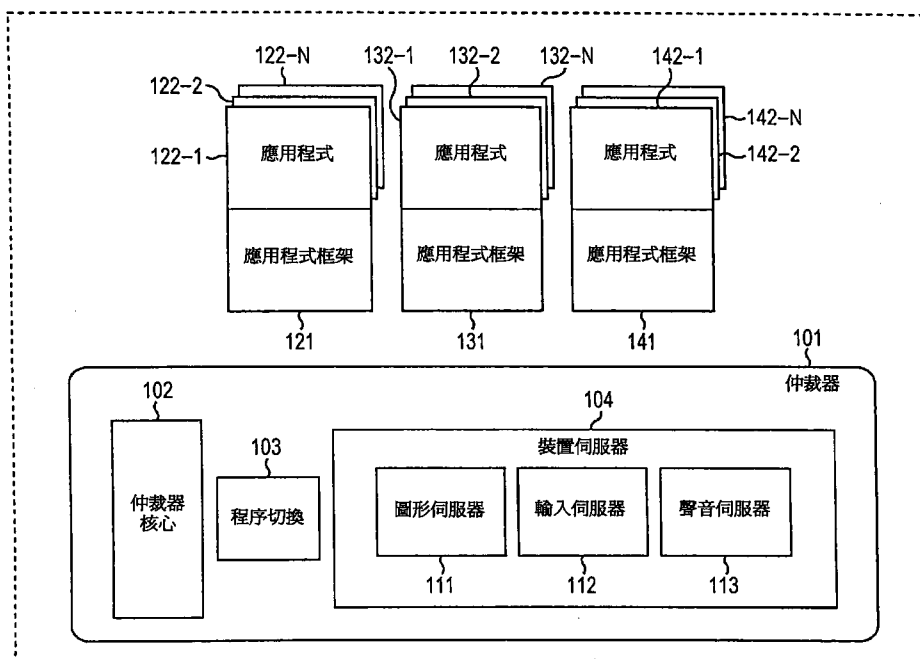
資訊處理裝置、資訊處理方法及程式

INFORMATION PROCESSING APPARATUS, INFORMATION PROCESSING METHOD, AND PROGRAM

(57)摘要

提出一種資訊處理裝置，包括複數個應用程式框架，在複數個應用程式框架上執行應用程式，以及一決定單元，配置以控制複數個應用程式框架之可操作狀態的切換。

第 3 圖



101：仲裁器

102：仲裁器核心

103：程序切換

104：裝置伺服器

111：圖形伺服器

112：輸入伺服器

113：聲音伺服器

121：應用程式框架

122-1-122-N：應用程式

131：應用程式框架

132-1-132-N：應用程式

141：應用程式框架

142-1-142-N：應用程式

發明摘要

※申請案號：102116202

※申請日：102 年 05 月 07 日

※IPC 分類：G06F 9/44 (2006.01)

G06F 9/48 (2006.01)

【發明名稱】(中文/英文)

資訊處理裝置、資訊處理方法及程式

Information processing apparatus, information processing method, and
program

【中文】

提出一種資訊處理裝置，包括複數個應用程式框架，在複數個應用程式框架上執行應用程式，以及一決定單元，配置以控制複數個應用程式框架之可操作狀態的切換。

【英文】

There is provided an information processing apparatus including a plurality of application frameworks upon which applications are executed, and a decision unit configured to control a switching of operable states of the plurality of application frameworks.

【代表圖】

【本案指定代表圖】：第(3)圖。

【本代表圖之符號簡單說明】：

101：仲裁器

102：仲裁器核心

103：程序切換

104：裝置伺服器

111：圖形伺服器

112：輸入伺服器

113：聲音伺服器

121：應用程式框架

122-1-122-N：應用程式

131：應用程式框架

132-1-132-N：應用程式

141：應用程式框架

142-1-142-N：應用程式

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：無

發明專利說明書

(本說明書格式、順序，請勿任意更動)

【發明名稱】(中文/英文)

資訊處理裝置、資訊處理方法及程式

Information processing apparatus, information processing method, and program

【技術領域】

本揭露關於一種資訊處理裝置、一種資訊處理方法、以及一種程式。具體來說，本揭露關於能夠進行用以切換複數個應用程式框架之程序的一種資訊處理裝置、一種資訊處理方法、以及一種程式。

【先前技術】

應用程式框架係為一種為了開發應用程式而提供有用函數的中間軟體群組。應用程式框架係基於一個應用程式框架存在於一個系統上的前提下而設計以便操作。因此，若使用者希望一個系統中使用複數個應用程式框架，則有需要藉由使用如當裝置電源關閉時之方法、應用程式框架能當其電源再次打開時改變之方法、提供單獨處理器和硬體以切換處理器和硬體之方法、或使用虛擬機之方法的方法，來切換應用程式框架。

PTL 1 揭露一種在使用將正在執行之應用程式轉變為單元來管理改變裝置之排除控制設定方面具有高擴展性的

電子裝置，以當擴展系統時解決困難。

[引用列表]

[專利文獻]

[PTL 1]

日本未經審查的專利申請書公開第 2008-210398 號

【發明內容】

[技術問題]

在一個系統上存在一個應用程式框架和允許應用程式操作在此應用程式框架上的情況下，必須設計嵌入應用程式框架的應用程式。因此，依據應用程式，可能難以設計以嵌入應用程式框架。於是，在開發應用程式和檢查操作方面的時間和成本增加。

希望能夠藉由切換複數個應用程式來使用操作在每個應用程式框架上的複數個應用程式。

[問題之解決方法]

根據本揭露之實施例之一種資訊處理裝置包括複數個應用程式框架，在複數個應用程式框架上執行應用程式、及一決定單元，配置以控制複數個應用程式框架之可操作狀態的切換。

根據本揭露之另一實施例之一種資訊處理方法包括執行關聯於一第一應用程式框架的一第一應用程式，切換一第二應用程式框架的一操作狀態以變成主動，及在切換第

二應用程式框架的操作狀態以變成主動之後執行關聯於第二應用程式框架的一第二應用程式。

根據本揭露之另一實施例，有提供一種以一程式實作的非暫態電腦可讀媒體，當程式被一電腦執行時使電腦進行一方法，方法包括執行關聯於一第一應用程式框架的一第一應用程式，切換一第二應用程式框架的一操作狀態以變成主動，及在切換第二應用程式框架的操作狀態以變成主動之後執行關聯於第二應用程式框架的一第二應用程式。

根據本揭露之實施例之一種資訊處理裝置包括複數個應用程式框架及一切換單元，切換應用程式框架。

根據本揭露之實施例之一種資訊處理裝置更包括複數個應用程式框架中的一第一應用程式框架、操作在第一應用程式框架上的一第一應用程式、不同於第一應用程式框架的一第二應用程式框架、及操作在第二應用程式框架上的一第二應用程式。當命令從第一應用程式切換為第二應用程式框架時，切換單元可將裝置的輸入和輸出目的地從第一應用程式框架切換為第二應用程式框架。

切換單元可進行切換以一對一地連接複數個應用程式框架和裝置。

切換單元可分別具有一個實現停止應用程式中的預定應用程式並啟動另一應用程式之函數的部分、及一個實現將複數個應用程式框架中的預定應用程式與裝置分開並將預定應用程式連接至另一應用程式框架之函數的部分。

當命令從第一應用程式切換為第二應用程式框架時，切換單元可停止第一應用程式並啟動第二應用程式。

第一應用程式可能有複數個，而切換單元可呈現為複數個第一應用程式中的一個應用程式，且切換單元為可從裝置傳送輸入給第二應用程式框架的第一應用程式。

切換單元可呈現為第一應用程式框架的一個功能，且包括在第一應用程式框架中的切換單元可從裝置傳送輸入給第二應用程式框架。

來自裝置的輸入可藉由供應包括在一作業系統中之函數的輸入，從第一應用程式或第一應用程式框架供應給第二應用程式框架。

供應函數的輸入可為 `uinput` 的函數。

來自第二應用程式的輸出資料可透過第二應用程式框架儲存在儲存單元中，且儲存在儲存單元中的輸出資料可被第一應用程式讀取而接著透過第一應用程式框架供應給裝置。

第二應用程式所畫的繪圖資料可透過第二應用程式儲存在儲存單元中，且儲存在儲存單元中的繪圖資料被第一應用程式讀取而接著透過第一應用程式框架供應給顯示器。

第一應用程式框架可為關於照相機功能的應用程式框架，且第二應用程式可為關於通訊功能的應用程式框架。

根據來自藉由第一應用程式操作在關於照相機功能之第一應用程式框架上的程序由使用者進行的輸入操作所擷

取之影像可藉由第二應用程式操作在關於通訊功能之第二應用程式框架上的程序傳送給另一裝置。

根據本揭露之另一實施例，一種資訊處理裝置的資訊處理方法，其中資訊處理裝置包括複數個應用程式框架，資訊處理方法包括切換應用程式框架。

根據本揭露之又一實施例，一種使電腦執行切換應用程式框架之程序的程式，其中電腦控制包括複數個應用程式框架的資訊處理裝置。

在資訊處理裝置中，提供了根據本揭露之實施例之資訊處理方法及程式、複數個應用程式框架並選擇性地切換複數個應用程式框架。

[本發明的有利效果]

根據本揭露之實施例，有可能切換複數個應用程式框架。此外，藉由切換複數個應用程式框架，有可能使用操作在每個應用程式框架上的複數個應用程式。

【圖式簡單說明】

第 1 圖係繪示依照實施例之資訊處理裝置之配置的圖。

第 2 圖係繪示依照實施例之應用程式框架的圖。

第 3 圖係繪示依照實施例之應用程式框架的圖。

第 4 圖係繪示依照實施例之應用程式框架之連接的圖。

第 5 圖係繪示依照實施例之關於應用程式框架之配置的圖。

第 6 圖係繪示依照實施例之切換應用程式框架的圖。

第 7 圖係繪示依照實施例之切換應用程式框架的圖。

第 8 圖係繪示依照實施例之切換應用程式框架的圖。

第 9 圖係繪示依照實施例之切換應用程式框架的圖。

第 10 圖係繪示依照實施例之切換應用程式框架的圖。

第 11 圖係繪示依照實施例之切換應用程式框架的圖。

第 12 圖係繪示依照實施例之切換應用程式框架的圖。

第 13 圖係繪示依照實施例之在輸入時之程序的圖。

第 14 圖係繪示依照實施例之在輸入時之程序的圖。

第 15 圖係繪示依照實施例之在輸入時之程序的圖。

第 16 圖係繪示依照實施例之在輸入時之程序的圖。

第 17 圖係繪示依照實施例之在輸入時之程序的圖。

第 18 圖係繪示依照實施例之在輸出時之程序的圖。

第 19 圖係繪示依照實施例之在輸出時之程序的圖。

【實施方式】

在下文中，將說明實現本揭露之實施例。此外，將如下來提出說明。

1. 資訊處理裝置配置

2. 軟體堆疊
3. 複數個應用程式框架之切換
4. 排除控制
5. 介面
6. 藉由仲裁器核心之應用程式框架的切換程序
7. 應用程式之切換
8. 仲裁器配置
9. 在啟動時的程序
10. 應用程式框架的切換程序
11. 關於在切換應用程式框架之後輸入的程序
12. 關於輸入的其他程序
13. 關於輸出的程序
14. 記錄媒體

[資訊處理裝置配置]

第 1 圖係繪示依照實施例之資訊處理裝置之配置的圖。本文中，資訊處理裝置 10 將被描述為一實例，且資訊處理裝置 10 可例如是個人電腦。此外，資訊處理裝置 10 可以是配置如數位相機的成像裝置或如電視機組的再生裝置之部分的裝置，且可建立於其中。

CPU（中央處理單元）21 接收一信號，其對應於由資訊處理裝置 10 之使用者透過輸入和輸出介面 25 和內部匯流排 24，使用輸入單元 26 所輸入的各種指令，並接著基於輸入信號來執行各種程序。

ROM（唯讀記憶體）22 基本上儲存程式中的固定資料和 CPU 21 所使用之計算的參數。RAM（隨機存取記憶體）23 儲存在執行 CPU 21 中使用的程式或資料和在其執行中隨需要改變的參數。CPU 21、ROM 22 和 RAM 23 係透過內部匯流排 24 而相互連接。

內部匯流排 24 亦耦接至輸入和輸出介面 25。輸入單元 26 可包括例如鍵盤、觸控板、微動滾輪、手寫板或滑鼠，且當資訊處理裝置 10 之使用者輸入各種指令給 CPU 21 時被操作。輸出單元 27 可包括例如如液晶顯示單元的顯示器並顯示各種為文字或影像形式的資訊。

儲存單元 28 可包括如 HDD（硬碟機）的儲存裝置，並記錄或再生 CPU 21 所執行的程式或資訊。通訊單元 29 係透過有線或無線通訊耦接至其他裝置，並接收和傳送資訊。如磁碟、光碟、磁光碟、或半導體記憶體的可移除媒體 31 依需求而裝設在驅動程式 30 上，並接收和傳送資料。

輸入單元 26 到驅動程式 30 的元件係透過輸入和輸出介面 25 和內部匯流排 24 耦接至 CPU 21。

[軟體堆疊]

接著，將說明實施例的操作環境。將藉由使用第 2 圖來說明藉由使用第 1 圖來說明的資訊處理裝置 10 之 CPU 21 所執行之在依照實施例之軟體程式中的軟體堆疊。在圖中，如圖之上部所示的模組被視為比如圖之下部所示的

模組更高層。

第 2 圖所示之軟體堆疊包括裝置 11-1 到 11-M、裝置驅動程式 12-1 到 12-M、作業系統 13、應用程式框架 14、及應用程式 15-1 到 15-N。

裝置 11-1 到 11-M 可以是例如如配置輸入單元 26 之鍵盤或滑鼠的硬體、和配置輸出單元 27 的顯示器。裝置驅動程式 12-1 到 12-M 係操作裝置 11-1 到 11-M 所需的軟體。本文中，認為裝置驅動程式 12-1 到 12-M 被供應作業系統 13，並接著將繼續敘述其之說明。另外，若沒有必要個別區分裝置 11-1 到 11-M 或裝置驅動程式 12-1 到 12-M，則它們能被簡單地描述為裝置 11 和裝置驅動程式 12。

作業系統 13 可以是例如提供待在各種應用程式 15 中共同使用之基本函數，並接著管理整個電腦系統的軟體，例如管理裝置驅動程式 12，其為操作如滑鼠、鍵盤之周邊裝置和配置輸入單元 26 和輸出單元 27 之顯示器的軟體，或管理儲存單元 28，其儲存各種資訊。

應用程式框架 14 可以是類別或程式庫的集合，其係用以實作程式化中的特定作業系統之應用程式的標準結構。由於每種平台或應用程式共同所需的代碼係事先提供在應用程式框架 14 中，因此當開發應用程式時，可只藉由在框架中整合和實作一特有函數來開發目標應用程式。

如第 2 圖所示，在依照實施例之一個系統中，存在一個應用程式框架 14，且存在嵌入應用程式框架 14 的應用

程式 15-1 到 15-N。應用程式框架 14 封裝裝置 11 並抽象化關於應用程式 15 的裝置，且通常只有一個應用程式框架 14 安裝在一個系統上。

若應用程式 15 適用於應用程式框架 14，則進行開發嵌入應用程式框架 14，藉此產生時間和成本。另一方面，這表示當開發應用程式時，不可能開發一目標應用程式而不在框架中整合和實作一特定函數。此外，根據至少如此，若應用程式 15 不適用於應用程式框架 14，則有一種可能性為進行待嵌入應用程式框架 14 之開發所導致的時間和成本浪費。

為了解決這項問題和其他情況，複數個應用程式框架 14 可設置在一個系統中，且開發適用於每個應用程式框架 14 的應用程式 15 以便實作。因此，容易增加新的應用程式（例如，函數）或之類。藉由切換複數個應用程式框架 14，可選擇性地使用位於應用程式框架 14 上的複數個應用程式 15，以便有助於實行函數或之類。

在下文中，將提出在複數個應用程式框架 14 安裝在一個系統中，選擇性地切換複數個應用程式框架 14，並接著可使用位於切換之應用程式框架 14 上的應用程式 15 之情況下的軟體堆疊配置之實例的說明。

[複數個應用程式框架之切換]

第 3 圖係繪示依照實施例之在複數個應用程式框架 14 安裝在一個系統中的情況下之軟體堆疊配置的圖。在

第 3 圖所示之軟體堆疊中，安裝了仲裁器 101、應用程式框架 121、應用程式 122-1 到 122-N、應用程式框架 131、應用程式 132-1 到 132-N、應用程式框架 141、及應用程式 142-1 到 142-N。

仲裁器 101 包括仲裁器核心 102、程序切換 103、及裝置伺服器 104。裝置伺服器 104 管理複數個裝置驅動程式，且在第 3 圖所示之實例中管理圖形伺服器 111、輸入伺服器 112、聲音伺服器 113 或之類。

此外，在第 3 圖所示之配置中，省略了第 2 圖所示之裝置 11 和作業系統 13 的圖。

應用程式 122-1 到 122-N 係為嵌入應用程式框架 121 的應用程式。同樣地，應用程式 132-1 到 132-N 係為嵌入應用程式框架 131 的應用程式，而應用程式 142-1 到 142-N 係為嵌入應用程式框架 141 的應用程式。

另外，本文中，應用程式 122、應用程式 132、和應用程式 142 的數量在圖中分別設成 N，但其數量可分別不同。又，在第 3 圖中，顯示出應用程式框架 121、應用程式框架 131、和應用程式框架 141 的三個應用程式框架，且將繼續敘述下面的說明。然而，這項說明並不表示其數量限於三個，而不用說，其數量可以是任何數量。

如第 3 圖所示，當允許應用程式框架 121、131 和 141 一起存在時，依據自己的應用程式框架之應用程式 122、132 和 142 可操作在一個系統中。此外，為了實現這點，可設置仲裁器 101。在下面的說明中，在應用程式

框架 121、131 和 141 之任一者設成目標的情況下，應用程式框架 121、131 和 141 被描述為應用程式框架。此外，在應用程式 122-1 到 122-N、132-1 到 132-N 和 142-1 到 142-N 之任一者設成目標的情況下，應用程式 122-1 到 122-N、132-1 到 132-N 和 142-1 到 142-N 被描述為應用程式。

應用程式框架 121 係為例如專用於照相機功能中的框架，以操作之前存在的照相機應用程式。例如，照相機功能可指出成像或再生靜止影像、成像或再生移動影像、變焦、自動聚焦或之類的功能。

應用程式框架 121 可以是專用於上述照相機功能中的框架，並提供開發數位相機應用程式所需的功能。應用程式 122 係專用於應用程式框架 121 之功能中的應用程式，並可以是實現照相機功能的應用程式。

然而，例如，應用程式框架 121 可能沒有透過網路通訊的通訊功能，但不同於應用程式框架 121 的應用程式框架 131 可能具有上述通訊功能。本文中，認為應用程式框架 131 具有透過網路通訊的功能。

應用程式框架 131 可以是實行智慧型手機中使用之通訊功能的應用程式框架，且具有瀏覽網際網路上之主頁、上載到雲端或之類的功能。

應用程式框架 131 上的應用程式 132 可設成用以上載擷取之影像至網路的應用程式，並藉由相對容易地使用應用程式框架 131 來開發上述應用程式 132。

由於應用程式框架 121 具有關於藉由使用數位相機或之類來成像的功能，因此應用程式框架 131 沒有必要具有關於如上述之照相機功能的功能。由此，有可能製造出專用於網路功能的應用程式框架 131。

彼此具有不同功能的上述應用程式框架會受到從在特定時刻正操作在系統上的一個應用程式框架切換為另一應用程式框架。例如，在特定時刻，應用程式框架 121 係可用的，而在特定時刻期間，使用應用程式框架 121 的應用程式 122 係處於可操作狀態（例如，正在操作）。此外，在其他特定時刻，應用程式框架 131 係處於可操作狀態。此應用程式框架的切換可由仲裁器 101 控制。來自應用程式的指令可進行此切換。

在實施例中，仲裁器 101 的仲裁器核心 102 基於來自啟動之應用程式的指令來判斷哪個應用程式框架可用。仲裁器 101 的仲裁器核心 102 基於判斷來控制切換至應用程式框架，並切換為操作在切換之應用程式框架上的應用程式。

仲裁器核心 102 進行一連串程序，例如命令操作中的應用程式框架停止之程序、命令可用的應用程式框架開始之程序、及通知使用系統資源之應用程式框架的裝置伺服器 104 之程序。

關於每個應用程式框架使用的裝置，裝置伺服器 104 進行哪個系統依據切換請求而耦接至裝置的排除程序。將參考第 4 圖來說明排除程序。另外，如參考第 5 圖所述，

裝置伺服器 104 在應用程式框架和系統資源（例如，硬體或之類的裝置）之間的介面中有所作用。

又，裝置伺服器 104 具有有關哪個應用程式目前可用的資訊，且具有如分配輸入裝置的資訊給目前可用的應用程式框架之功能的功能，以及只從目前可用之應用程式框架輸出資訊作為輸出裝置之資訊的功能。

再者，如參考第 6 圖所述，在仲裁器 101 中，仲裁器核心 102 也可以是基於來自應用程式的請求來呼叫程序切換 103 和裝置伺服器 104 並切換應用程式框架的模組。

如參考第 7 圖所述，程序切換 103 可以是啟動或停止執行每個應用程式框架的應用程式之程序並切換應用程式之執行物件的模組。

本文中，假設仲裁器 101 有設置仲裁器核心 102、程序切換 103 和裝置伺服器 104，且將繼續敘述其之說明。然而，仲裁器 101 可控制選擇性地切換複數個應用程式框架，而仲裁器 101 的配置並不限於仲裁器 102、程序切換 103 和裝置伺服器 104 之各者被單獨設置的配置，如第 3 圖所示。

[排除控制]

首先，參考第 4 圖，將依照實施例來說明由裝置伺服器 104 進行且哪個系統係依照切換請求耦接至裝置的排除程序。關於如鍵盤的裝置，由於系統中通常存在一個裝置，因此兩個應用程式框架不可能同時使用裝置。因此，

必須進行排除程序，其中一個應用程式框架係耦接至一個裝置，使得二或更多應用程式框架不會同時使用一個裝置。

如第 4A 圖所示，假設應用程式框架 121 可耦接至輸入伺服器 112，而應用程式框架 141 可耦接至圖形伺服器 111。

此外，當應用程式框架 121 被視為是客戶時，圖形伺服器 111 和輸入伺服器 112 被命名為伺服器，因為圖形伺服器 111 和輸入伺服器 112 實現伺服器的作用。例如，如之後所述，作為伺服器的圖形伺服器 111 具有提供資料給圖形裝置驅動程式 161 的介面功能，而作為客戶的應用程式框架 121 可藉由使用圖形伺服器 111 的功能來供應資料給圖形裝置驅動程式 161。

由於上述伺服器與客戶之間的關係存在，因此，本文中，使用敘述「伺服器」以類似方式來稱為圖形伺服器 111 和輸入伺服器 112。

在如第 4A 圖所示之狀態下，應用程式框架 121 係一對一地耦接至輸入伺服器 112，而應用程式框架 141 係一對一地耦接至圖形伺服器 111。因此，正常地進行排除程序。

同樣地，如第 4B 圖所示，可假設應用程式框架 131 係耦接至圖形伺服器 111，而應用程式框架 141 可耦接至輸入伺服器 112。在這種情況下，應用程式框架 131 係一對一地耦接至圖形伺服器 111，而應用程式框架 141 係一

對一地耦接至輸入伺服器 112。因此，正常地進行排除程序。

以上述方式，裝置伺服器 104 進行排除控制，使得應用程式框架係一對一地耦接至裝置伺服器 104 中的裝置伺服器。

[介面]

接下來，將參考第 5 圖來說明依照實施例之應用程式框架和系統資源之間的介面。如第 5 圖所示，藉由裝置伺服器 104 中的圖形伺服器 111 和圖形裝置驅動程式 161 之間的圖形裝置介面 151 來居間進行交換資料。圖形裝置驅動程式 161 係為控制輸出資料給顯示器 171 的驅動程式。

藉由裝置伺服器 104 中的輸入伺服器 112 和輸入裝置驅動程式 162 之間的輸入裝置介面 152 來居間進行交換資料。輸入裝置驅動程式 162 係為控制來自鍵盤 172 或滑鼠 173 之輸入的驅動程式。雖然未顯示在圖中，但聲音伺服器 113 透過介面與驅動程式交換資料。

應用程式框架對圖形裝置驅動程式 161 和輸入裝置驅動程式 162 的介面使用圖形伺服器 111 和裝置伺服器 104 中的輸入伺服器 112。也就是說，認為裝置伺服器 104 具有介面功能，而由此裝置驅動程式 104 具有將驅動程式對應於應用程式框架的功能。

[藉由仲裁器核心之應用程式框架的切換程序]

然後，將參考第 6A 和 6B 圖來說明依照實施例之藉由仲裁器核心 102 之應用程式框架的切換。第 6A 圖繪示在特定時間的連接狀態。應用程式框架 121 係耦接至圖形伺服器 111，且應用程式框架 121 係耦接至輸入伺服器 112。在這種狀態下，對仲裁器核心 102 命令從應用程式 122 切換為應用程式框架 131。

在下一個時段，接收上述指令的仲裁器核心 102 對裝置伺服器 104 發出切換至應用程式框架 131 的指令。當裝置伺服器 104 從仲裁器核心 102 接收指令時，裝置伺服器 104 基於指令將圖形伺服器 111 和輸入伺服器 112 的連線目的地從應用程式框架 121 切換至應用程式框架 131，如第 6B 圖所示。

以上述方式，仲裁器核心 102 依據來自應用程式的指令，發出將應用程式框架切換為裝置伺服器 104 的指令。裝置伺服器 104 基於來自仲裁器核心 102 的指令，進行切換連線目的地的應用程式框架。

[應用程式之切換]

儘管進行如上述的應用程式框架之切換，仍也必須進行應用程式之切換，以便由切換之目的地的應用程式開始程序。將參考第 7A 和 7B 圖來說明依照實施例的應用程式之切換。

第 7A 圖繪示應用程式框架 121 操作及應用程式框架 121 上之應用程式 122 操作的狀態。在這種狀態下，從應

應用程式 122 對仲裁器核心 102 發出切換為應用程式框架 131 的指令。

當仲裁器核心 102 從應用程式 122 接收指令時，仲裁器核心 102 基於指令對程序切換 103 發出切換為應用程式 132 的指令，如第 7B 圖所示。程序切換 103 辨識出正當從仲裁器核心 102 接收指令時操作的應用程式，並對識別之應用程式發出停止操作的指令。在這種情況下，程序切換 103 對應用程式 122 發出停止操作的指令。應用程式 122 基於指令來停止操作。

另一方面，程序切換 103 對應用程式 132 發出開始操作的指令。應用程式 132 基於指令來開始操作。

以上述方式，可由程序切換 103 切換應用程式。此外，可從仲裁器核心 102 發出將應用程式切換為程序切換 103 的指令。如上所述，仲裁器核心 102 也可對裝置伺服器 104 發出切換連線目的地之應用程式框架的指令。

以此方式，程序切換 103 可基於來自仲裁器核心 102 的指令來切換應用程式，而裝置伺服器 104 可切換應用程式框架。於是，有可能切換應用程式框架並切換操作在應用程式框架上的應用程式。

[仲裁器配置]

依照實施例之仲裁器 101 的配置和仲裁器 101 與其他子系統之間的關係係繪示在第 8 和 9 圖中。參考第 8 和 9 圖，仲裁器 101 的仲裁器核心 102 從應用程式 122 和應用

程式 132 接收指令。另外，仲裁器核心 102 係耦接至程序切換 103 以對其發出指令，並也耦接至裝置伺服器 104 中的每個裝置伺服器。

如第 9 圖所示，配置依照實施例之裝置伺服器 104 具有一開關。在第 9 圖所示之實例中，執行輸入裝置驅動程式 162 和應用程式框架 121 或應用程式框架 131 之間連接切換的開關係顯示在圖中。雖然未顯示在圖中，但執行圖形裝置驅動程式 161 和應用程式框架 121 或應用程式框架 131 之間連接切換的開關或之類也包括在裝置伺服器 104 中。

如上所述，裝置伺服器 104 可配置為具有複數個開關，其切換每個裝置驅動程式和應用程式框架之間的連接。

[在啓動時的程序]

接著，依照實施例和參考如第 10 圖所示之序列圖，將在具有進行以上參考第 3 圖到第 9 圖所述之應用程式框架的切換之軟體堆疊的系統（或裝置）中敘述待在啓動時執行的程序。

例如，當第 1 圖所示之資訊處理裝置 10 的電源開啓時，從 CPU 21 對仲裁器核心 102 或之類發出初始化命令。在步驟 S 11 中，仲裁器核心 102 進行初始化程序。

在步驟 S 51 中，圖形伺服器 111 也進行初始化程序，且在完成初始化之後，對仲裁器核心 102 提出完成初

始化的通知。在步驟 S 12 中，仲裁器核心 102 接收通知。

同樣地，在步驟 S 71 中，輸入伺服器 112 也進行初始化程序，且在完成初始化之後，對仲裁器核心 102 提出完成初始化的通知。在步驟 S 13 中，仲裁器核心 102 接收通知。由裝置伺服器 104 中的每個裝置伺服器執行如上述之程序。如第 10 圖所示且在下面的說明中敘述，將示範和說明依照實施例之圖形伺服器 111 和輸入伺服器 112。

同樣地，在步驟 S 91 中，程序切換 103 也進行初始化程序，且在完成初始化之後，對仲裁器核心 102 提出完成初始化的通知。在步驟 S 14 中，仲裁器核心 102 接收通知。以此方式，仲裁器核心 102 接收完成初始化的通知而由此辨識出裝置伺服器 104 中之裝置伺服器的狀態和程序切換 103 的狀態。

本文中，將示範和說明由 CPU 21 命令啟動應用程式 132 的情況。在這種情況下，啟動應用程式框架 131 和應用程式 132。當啟動應用程式 132 時，在步驟 S 101 中，與裝置連接的請求從應用程式 132 轉送給仲裁器核心 102。

在步驟 S 15 中，當仲裁器核心 102 從應用程式 131 接收請求時，在仲裁器核心 102 和圖形伺服器 111 之間執行回呼程序（步驟 S 16 和步驟 S 52）。此外，也在仲裁器核心 102 和輸入伺服器 112 之間執行回呼程序（步驟 S

17 和步驟 S 72)。

如以上關於第 6 圖所述且依照實施例，藉由裝置伺服器 104 的排除控制，應用程式框架和裝置伺服器（例如，特定裝置）係一對一地耦接。在這種情況下，圖形伺服器 111 和輸入伺服器 112 分別耦接至應用程式框架 131。當完成上述連接時，分別從圖形伺服器 111 和輸入伺服器 112 提出完成連接的通知（步驟 S 53 和步驟 S 73）。

在步驟 S 18 和步驟 S 19 中，仲裁器核心 102 從圖形伺服器 111 和輸入伺服器 112 接收通知。在步驟 S 20 中，完成與裝置之連接的通知被提出給應用程式框架 131（或應用程式 132），且在步驟 S 102 中，被應用程式 132 接收。當完成上述連接時，開始應用程式 132 所進行的程序。

[應用程式框架的切換程序]

參考第 11 圖，將說明依照實施例之應用程式框架的切換。本文中，將示範和說明啟動應用程式框架 131 和應用程式 132 和進行切換為應用程式框架 121 的情況。

在步驟 S 201 中，從應用程式 132 轉送對切換為應用程式框架 121 的請求給仲裁器核心 102。在步驟 S 221 中接收請求的仲裁器核心 102 在步驟 S 222 中進行與圖形伺服器 111 的回呼程序。圖形伺服器 111 進行與仲裁器核心 102 的回呼程序作為步驟 S 241 中的程序。

同樣地，仲裁器核心 102 在步驟 S 223 中進行與輸入

伺服器 112 的回呼程序。輸入伺服器 112 進行與仲裁器核心 102 的回呼程序作為步驟 S 261 中的程序。

如上述進行之回呼程序係為釋放與應用程式框架 131 連接的程序。當如上所述地進行釋放連接時，分別從圖形伺服器 111 和輸入伺服器 112 提出指出完成釋放連接的通知（步驟 S 242 和步驟 S 262）。

在步驟 S 224 和步驟 S 225 中，當仲裁器核心 102 從圖形伺服器 111 和輸入伺服器 112 接收通知時，在步驟 S 226 中，仲裁器核心 102 通知應用程式框架 131 釋放裝置和應用程式框架 131 之間的連接。在步驟 S 202 中，應用程式框架 131 如上所述地接收通知並因此辨識出釋放與裝置伺服器的連接。

在步驟 S 227 中，儘管仲裁器核心 102 對應用程式框架 131 提出上述通知，但仲裁器核心 102 對程序切換 103 發出從應用程式 132 切換為應用程式 122 的指令。

在步驟 S 281 中，從仲裁器核心 102 接收指令的程序切換 103 命令應用程式 132 停止步驟 S 282 中的操作（亦即，執行應用程式框架 131 的停止程序）。

此外，在步驟 S 283 中，程序切換 103 命令應用程式 122 開始操作（亦即，執行應用程式框架 121 的啟動程序）。以上述方式，進行應用程式的切換。在此例中，進行從應用程式 132 切換為應用程式 122。

當由來自如上述之程序切換 103 的指令進行應用程式的切換時，從程序切換 103 對仲裁器核心 102 提出完成應

用程式之切換的通知（步驟 S 284 中）。在步驟 S 228 中，仲裁器核心 102 藉由來自程序切換 103 的通知，辨識出此時正在操作的應用程式，在此例中是應用程式 122。

在步驟 S 229 中，仲裁器核心 102 進行與圖形伺服器 111 的回呼程序，且在步驟 S 230 中，進行與輸入伺服器 112 的回呼程序。回呼程序係為分別耦接應用程式框架 121 和圖形伺服器 111，以及耦接應用程式框架 121 和輸入伺服器 112 的程序。

當執行如上述之連接程序時，指出分別從圖形伺服器 111 和輸入伺服器 112 提出指出完成連接的通知（步驟 S 244 和步驟 S 264）。在步驟 S 231 和步驟 S 232 中，當仲裁器核心 102 從圖形伺服器 111 和輸入伺服器 112 接收通知時，在步驟 S 233 中，仲裁器核心 102 通知應用程式框架 121 完成與裝置之連接。以上述方式，進行應用程式框架的切換。

參考第 12 圖，將再次說明依照實施例之應用程式框架的切換。本文中，將示範和說明啟動應用程式框架 121 和應用程式 122 和進行切換為應用程式框架 131 的情況。

在步驟 S 301 中，從應用程式 122 轉送對切換為應用程式框架 131 的請求給仲裁器核心 102。在步驟 S 331 中接收如上述之請求的仲裁器核心 102 在步驟 S 332 中進行與圖形伺服器 111 的回呼程序。圖形伺服器 111 進行與仲裁器核心 102 的回呼程序作為步驟 S 341 中的程序。

同樣地，在步驟 S 333 中，仲裁器核心 102 進行與輸

入伺服器 112 的回呼程序。輸入伺服器 112 進行與仲裁器核心 102 的回呼程序作為步驟 S 361 中的程序。

如上述進行之回呼程序係為釋放與應用程式框架 121 連接的程序。當如上所述地進行釋放連接時，分別從圖形伺服器 111 和輸入伺服器 112 提出指出完成釋放連接的通知（步驟 S 342 和步驟 S 362）。

在步驟 S 334 和步驟 S 335 中，當仲裁器核心 102 從圖形伺服器 111 和輸入伺服器 112 接收通知時，在步驟 S 336 中，仲裁器核心 102 通知應用程式框架 121 釋放與裝置的連接。儘管仲裁器核心 102 提出上述通知，但在步驟 S 337 中，仲裁器核心 102 對程序切換 103 發出從應用程式 122 切換為應用程式 132 的指令。

在步驟 S 381 中，從仲裁器核心 102 接收指令的程序切換 103 命令應用程式 122 停止步驟 S 382 中的操作（亦即，執行應用程式框架 121 的停止程序）。

此外，在步驟 S 383 中，程序切換 103 命令應用程式 132 開始操作（亦即，執行應用程式框架 131 的啟動程序）。以上述方式，進行應用程式的切換。在此例中，進行從應用程式 122 切換為應用程式 132。

當由來自如上述之程序切換 103 的指令進行應用程式的切換時，從程序切換 103 對仲裁器核心 102 提出完成應用程式之切換的通知（步驟 S 384 中）。在步驟 S 338 中，仲裁器核心 102 藉由來自程序切換 103 的通知，辨識出此時正在操作的應用程式，在此例中是應用程式 132。

在步驟 S 339 中，仲裁器核心 102 進行與圖形伺服器 111 的回呼程序，且在步驟 S 340 中，進行與輸入伺服器 112 的回呼程序。回呼程序係為分別耦接應用程式框架 131 和圖形伺服器 111，以及耦接應用程式框架 131 和輸入伺服器 112 的程序。

當執行如上述之連接程序時，分別從圖形伺服器 111 和輸入伺服器 112 提出指出完成連接的通知（步驟 S 344 和步驟 S 364）。在步驟 S 341 和步驟 S 342 中，當仲裁器核心 102 從圖形伺服器 111 和輸入伺服器 112 接收通知時，在步驟 S 343 中，仲裁器核心 102 通知應用程式框架 131 完成與裝置之連接。以上述方式，進行應用程式框架的切換。

以上述方式，可在一個系統中設置複數個應用程式框架。因此，有可能切換複數個應用程式框架。可在高速下進行複數個應用程式框架的切換，而不進行如裝置電源關閉時，而接著將其電源再次打開之方法、提供單獨處理器和硬體以切換處理器和硬體之方法、或使用虛擬機之方法的方法。

另外，在作業系統的使用期限中，有可能在一個系統中使用依據各種應用程式框架的應用程式。此外，儘管多樣地使用過去的應用程式框架和應用程式，但仍有可能將對應於不同於過去的應用程式框架之應用程式框架的應用程式嵌入系統。這變成有可能的，而因此有可能降低應用程式的開發成本。

[在切換應用程式框架之後的程序]

接下來，將說明如上述之切換應用程式框架之後的程序。首先，爲了參考之目的，將參考第 13 圖來說明不可能進行如上述之切換應用程式框架的情況。

參考第 13 圖，來自作爲輸入裝置之裝置 11（例如鍵盤 172 和滑鼠 173）的輸入資料係透過作業系統 13 上的裝置驅動程式 12 供應給應用程式框架 14。此外，輸入資料係透過應用程式框架 14 供應給應用程式 15，且由應用程式 15 的程序執行對應於輸入資料的程序。

如之前參考第 2 圖所述，一個系統係基於在其中只出現一個應用程式框架 14 操作的前提下而設計。於是，例如，如第 13 圖所示，當在一個系統上出現不同於應用程式框架 14 的應用程式框架 14'或應用程式框架 14"時，沒有任何機制從裝置 11 供應輸入資料給應用程式框架 14'或應用程式框架 14"。因此，不可能對其供應輸入資料。

此外，依據應用程式框架 14'或應用程式框架 14"的應用程式 15'或應用程式 15"可能不會基於輸入資料來執行程序。換言之，只有應用程式框架 14 上的應用程式 15 可使用來自輸入裝置的輸入資料。

然而，如上所述，根據實施例，在一個系統中設置複數個應用程式框架，且接著可藉由切換複數個應用程式框架來使用複數個應用程式框架。下面將說明依照實施例的程序，其中切換應用程式框架，且接著傳送來自輸入裝置

的輸入資料給應用程式。

第 14 圖係繪示依照實施例之切換應用程式框架，且接著傳送來自輸入裝置的輸入資料給應用程式之配置實例的圖。在第 14 圖所示之實例中，為接收輸入之應用程式的重分配器 201 可被單獨地提供應用程式 122-1 到 122-N。

再者，為了從輸入裝置供應輸入資料給在複數個應用程式框架中操作的應用程式框架，可使用作業系統 13，例如，Linux（註冊商標）的 UINPUT 函數。作業系統 13 的 UINPUT 函數可依據來自應用程式的指令來產生任意的輸入事件。亦即，UINPUT 函數具有供應應用程式之指令所輸入的資料給其他函數的函數。

藉由使用這項函數，增加進行「當接收一輸入事件時，傳送輸入事件給 UINPUT」之程序的模組作為應用程式框架上的應用程式。此模組可以是如第 14 圖所示之重分配器 201。輸入事件係由重分配器 201 傳送給 uinput 211。因此，其他應用程式框架 131 和應用程式框架 141 可接收輸入事件。

如第 14 圖所示，當應用程式框架 131 正在操作時，首先，來自作為輸入裝置之裝置 11（例如鍵盤 172 和滑鼠 173）的輸入資料係透過作業系統 13 上的裝置驅動程式 12 供應給應用程式框架 121。此外，輸入資料係透過應用程式框架 121 供應給重分配器 201，且藉由重分配器 201 的程序供應給作業系統 13 的 uinput 211。

輸入資料係由 uinput 211 的程序供應給應用程式框架 131。以上述方式，當供應輸入資料給重分配器 201 時，由重分配器 201 的程序將輸入資料回傳給作業系統 13 的 uinput 211。之後，由 uinput 211 的程序供應輸入資料給此時正在操作的應用程式框架。

如上所述，重分配器 201 具有透過作業系統 13 從輸入裝置供應輸入資料給應用程式框架的功能。這項功能係類似於裝置伺服器 104 的功能。例如，如以上參考第 5 圖所述，裝置伺服器 104 具有從裝置供應資料給應用程式框架的介面功能。

藉此，第 14 圖所示之配置可被視為是在重分配器 201 中設置裝置伺服器 104 的情況。以此方式，裝置伺服器 104 可設置作為重分配器 201。換言之，裝置伺服器 104 可設置作為應用程式。另外，沒有必要完全獨立地提供裝置伺服器 104，而裝置伺服器 104 可配置以共享其他功能。

將參考第 15 圖之序列圖來說明涉及在如第 14 圖所示之配置中輸入的程序。在步驟 S 501 中，從裝置 11 供應輸入資料供應給裝置驅動程式 12。裝置驅動程式 12 從裝置 11 供應輸入資料給應用程式框架 121（步驟 S 502）。

在步驟 S 521 中，應用程式框架 121 從裝置驅動程式 12 接收輸入資料。如上所述，當在應用程式框架 121 中產生輸入事件時，在步驟 S 522 中，應用程式框架 121 供應輸入資料給重分配器 201。

如上所述，重分配器 201 係為進行當收到輸入事件時，傳送輸入事件給 UINPUT 之程序的應用程式。由此，在步驟 S 541 中，當重分配器 201 從應用程式框架 121 接收輸入資料之供應時，在步驟 S 542 中，重分配器 201 將輸入資料回傳給 uinput 211。

在步驟 S 561 中，當 uinput 211 從重分配器 201 接收輸入資料時，在步驟 S 562 中，uinput 211 供應輸入資料給應用程式框架 131。在這種情況下，由於應用程式框架 131 係藉由仲裁器核心 102 之程序而對應於裝置，因此供應輸入資料給應用程式框架 131。例如，當應用程式框架 141 係藉由仲裁器核心 102 之程序而對應於裝置時，供應輸入資料給應用程式框架 141。

在步驟 S 581 中，應用程式框架 131 從 uinput 211 接收輸入資料，在應用程式框架 131 中產生輸入事件。當在應用程式框架 131 中產生輸入事件時，在步驟 S 582 中，應用程式框架 131 供應收到之輸入資料給高層應用程式 132。在步驟 S 591 中，當應用程式 132 接收輸入資料時，應用程式 132 基於此輸入資料來開始程序。

如上所述，設置重分配器 201，從重分配器 201 供應輸入資料給 uinput 211，且接著允許從 uinput 211 傳送輸入資料給此時正在操作的應用程式框架。因此，有可能從如鍵盤和滑鼠的輸入裝置供應輸入給已切換的應用程式框架。

[關於輸入的其他程序]

將參考第 16 圖來說明依照實施例之在切換應用程式框架之後傳送輸入裝置的輸入資料給應用程式之情況的另一配置實例。第 16 圖所示之實例也使用作業系統 13，例如，Linux（註冊商標）的 UINPUT 函數以便以類似於第 14 圖所示之配置實例的方式，從輸入裝置供應輸入資料給複數個應用程式框架中正在操作的應用程式框架。

在第 16 圖所示之配置中，在應用程式框架 301 中設置輸入和輸出決定單元 302，且由輸入和輸出決定單元 302 決定輸入之輸入資料的輸出目的地，藉此供應輸入資料給決定之目的地。輸入和輸出決定單元 302 判斷透過裝置驅動程式 12 輸入的輸入資料是對應用程式 122 的輸入資料或對有別於應用程式 122 之應用程式的輸入資料。於是，基於此判斷供應輸入資料給應用程式。

在第 16 圖所示之配置中，在應用程式框架 301 中設置的輸入和輸出決定單元 302 具有透過作業系統 13（uinput 211）從輸入裝置供應輸入資料給應用程式框架的功能。這項功能係類似於裝置伺服器 104 的功能。例如，如以上參考第 5 圖所述，裝置伺服器 104 具有從裝置供應輸入資料給應用程式框架的介面功能。

藉此，第 16 圖所示之配置可被視為是在應用程式框架 301 中設置裝置伺服器 104 的情況。如上所述，有可能提供裝置伺服器 104 作為應用程式框架的一個功能。換言之，有可能提供裝置伺服器 104 作為應用程式框架。另

外，沒有必要完全獨立地提供裝置伺服器 104，而裝置伺服器 104 可配置以共享其他功能。

將參考第 17 圖之序列圖來說明涉及依照實施例輸入的程序。在步驟 S 701 中，當進行從裝置 11 輸入至裝置驅動程式 12 時，在步驟 S 702 中，裝置驅動程式 12 供應輸入資料給應用程式框架 301。在步驟 S 711 中，應用程式框架 301 的輸入和輸出決定單元 302 判斷來自裝置驅動程式 12 的輸入資料是輸出至應用程式 122 或輸出至有別於應用程式 122 的應用程式。

由於輸入和輸出決定單元 302 具有裝置伺服器 104 的功能，因此輸入和輸出決定單元 302 管理裝置和應用程式框架之間的關聯。輸入和輸出決定單元 302 基於管理之關聯來決定輸入資料的供應目的地。

當輸入和輸出決定單元 302 判斷輸出輸入資料至應用程式 122 時，在步驟 S 712 中，輸入和輸出決定單元 302 輸出輸入資料給應用程式 122。在步驟 S 721 中，當應用程式 122 從輸入和輸出決定單元 302 接收輸入資料時，應用程式 122 基於此輸入資料來開始程序。

另一方面，當輸入和輸出決定單元 302 判斷輸出輸入資料至有別於應用程式 122 的應用程式時，在步驟 S 713 中，輸入和輸出決定單元 302 輸出輸入資料給 uinput 211。此外，在第 17 圖中，顯示出在步驟 S 712 中，從輸入和輸出決定單元 302 供應輸入資料給應用程式 122 的情況，以及在步驟 S 713 中，從輸入和輸出決定單元 302 供

應輸入資料給 uinput 211 的情況。然而，實際上，基於輸入和輸出決定單元 302 的判斷只可進行其中一個供應。

在步驟 S 741 中，當 uinput 211 從輸入和輸出決定單元 302 接收輸入資料時，在步驟 S 742 中，uinput 211 輸出輸入資料給此時正在操作的應用程式框架 131。因此，在步驟 S 751 中，應用程式框架 131 接收輸入資料。

以此方式，當在應用程式框架 131 中產生輸入事件時，在步驟 S 752 中，應用程式框架 131 將供應之輸入資料供應給高層應用程式 312。在步驟 S 761 中，當應用程式 132 從應用程式框架 131 接收輸入資料時，應用程式 132 基於此輸入資料來開始程序。

以上述方式，有可能從輸入裝置供應輸入資料給所欲之應用程式（亦即，應用程式框架）。

如上所述，有可能從輸入裝置供應輸入資料給所欲之應用程式（亦即，應用程式框架）。因此，即使在必須改變預定應用程式框架的情況下，其改變量仍很小，那麼整個系統可容易地擴展。此外，有可能進行特殊程序，例如在分配之前校正值或過濾的程序、或同時傳送輸入事件給複數個應用程式框架的程序，以藉由應用程式或應用程式框架來重分配輸入事件。例如，移除不必要的事件並校正指標座標。由此，有可能進行決定應用程式框架的程序，其基於指標座標或之類來分配事件。

[關於輸出的程序]

然後，將說明在切換應用程式框架之後，從預定應用程式傳送資料給輸出裝置的情況。如第 18 圖所示且依照實施例，可藉由提供共享記憶體 401 來實現輸出。

共享記憶體 401 係為在應用程式 122 和應用程式框架 131 中共享和使用的記憶體。另外，雖然未顯示在第 18 圖中，但共享記憶體 401 也是可在應用程式 122 和應用程式框架 141 中共享和使用的記憶體。也就是說，共享記憶體 401 可以是在預定應用程式之間共享的記憶體，在此例中，是應用程式框架 121 上的應用程式 122，以及不同於預定應用程式之應用程式的低層應用程式框架。

應用程式框架 131（亦即，應用程式框架 131 的高層應用程式 132）所畫的資料係一次儲存在共享記憶體 401 中。然後，儲存在共享記憶體 401 中的繪圖資料係由應用程式 122 讀取而接著供應給應用程式框架 121。

之後，繪圖資料藉由顯示文字或如影像之圖形（由應用程式 132 所畫）在顯示器 171 上的應用程式框架 121，透過裝置驅動程式 12 供應給裝置 11。

將參考第 19 圖之序列圖來說明依照實施例且如上所述之程序。關於第 19 圖所說明之在輸出時的程序係類似於第 18 圖所示之情況。將示範和顯示佔用圖形裝置的應用程式框架 121 和應用程式 132 所畫的圖形係顯示在顯示器 171 上的情況。

在步驟 S 901 中，應用程式 132 開始繪圖以輸出繪圖資料至應用程式框架 131。在步驟 S 921 中，應用程式框

架 131 將收到之繪圖資料寫入共享記憶體 401。以此方式，來自應用程式 132 的繪圖資料被寫入共享記憶體 401。

應用程式 122 具有當完成由應用程式框架 131 將繪圖資料寫入共享記憶體 401 時，從應用程式框架 131 接收通知，並讀取寫入在共享記憶體 401 中之繪圖資料的功能。

當應用程式 122 從應用程式框架 131 接收通知時，在步驟 S 941 中，應用程式 122 從共享記憶體 401 讀取繪圖資料。讀取繪圖資料係由應用程式 122 的程序供應給應用程式框架 121。當應用程式框架 121 如上在步驟 S 961 中所述地接收供應時，在步驟 S 962 中，應用程式框架 121 供應繪圖資料給作業系統 13（例如，裝置驅動程式 12）。

之後，從裝置驅動程式 12 供應繪圖資料給裝置 11（例如，顯示器 171）。因此，應用程式 132 所畫的圖形會顯示在顯示器 171 上。

如上所述，共享記憶體 401 係由應用程式框架 131 和應用程式 122 共享和使用。應用程式 122 從共享記憶體 401 讀取繪圖資料以供應繪圖資料給裝置。由此，進行繪圖。也可以類似方式透過共享記憶體 401 來進行如音頻資料的其他輸出。

以上述方式，即使在複數個應用程式框架中的預定應用程式框架可能不直接供應繪圖資料給裝置 11（例如，顯示器 171）的配置中，仍可由另一應用程式框架和另一

應用程式框架上之應用程式的程序從預定應用程式框架供應繪圖資料給裝置 11。由於有可能進行上述程序，因此應用程式框架改變較少而應用程式框架增加。因此，有可能擴展整個系統。

[記錄媒體]

可由硬體執行或可由軟體執行上述一連串程序。在由軟體執行一連串程序的情況下，配置軟體的程式係安裝在電腦上。在此，電腦之實例包括例如於其中內建專用硬體的電腦、通用個人電腦（其能夠藉由安裝各種程式來執行各種功能）、等等。

例如，在第 1 圖所示之資訊處理裝置 10 中，藉由執行儲存在儲存單元 28 中之程式的 CPU 21，例如，藉由經由輸入和輸出介面 25 和內部匯流排 24 將程式載入 RAM 23 來進行上述一連串處理。

例如，藉由記錄在可移除媒體 31 上作為封裝媒體或之類，能夠提供電腦（例如，CPU 21）執行的程式。再者，能夠經由如區域網路、網際網路、或數位衛星廣播的有線或無線傳輸媒體來提供程式。

在電腦中，程式能夠藉由將可移除媒體 31 裝到驅動程式 30，經由輸入和輸出介面 25 來安裝在儲存單元 28 上。另外，程式能夠藉由經由有線或無線傳輸媒體被通訊單元 29 接收來安裝在儲存單元 28 上。另一方面，程式也可事先安裝在 ROM 22 或儲存單元 28 上。

此外，電腦所執行的程式可以是沿著本說明書所述之順序在時間序列中進行處理的程式，或可以是在如平行或當進行呼叫時之必要時機進行處理的程式。

另外，在本說明書中，系統可表示複數個裝置所形成的整個裝置。

再者，在本說明書中，「連接」和「耦接」之術語可意味著元件係直接連接或耦接在一起，或可透過一或更多中間元件來連接或耦接。

又，本揭露之實施例並不限於上述實施例，且在不脫離本揭露之主旨下可能進行各種修改。

另外，本揭露也可採用下列配置。

(1) 一種資訊處理裝置，包括：

複數個應用程式框架，在複數個應用程式框架上執行應用程式；及

一決定單元，配置以控制複數個應用程式框架之可操作狀態的切換。

(2) 如第(1)項所述之資訊處理裝置，其中決定單元基於來自在複數個應用程式框架中之選擇性的一個上執行之一應用程式的指令來控制複數個應用程式框架之可操作狀態的切換。

(3) 如第(2)項所述之資訊處理裝置，更包括：

一程序切換單元，配置以切換一主動應用程式為在複數個應用程式框架中之選擇性的一個上執行之應用程式。

- (4) 如第(3)項所述之資訊處理裝置，其中程序切換單元更配置以停止一目前啟動的應用程式之操作並開始在複數個應用程式框架中之選擇性的一個上執行之應用程式的操作。
- (5) 如第(1)項所述之資訊處理裝置，其中決定單元更配置以供應一輸入資料給複數個應用程式框架中之選擇性的一個。
- (6) 如第(5)項所述之資訊處理裝置，其中決定單元更配置以基於來自在複數個應用程式框架中之選擇性的一個上執行之一應用程式的指令來供應輸入資料給複數個應用程式框架中之選擇性的一個。
- (7) 如第(5)項所述之資訊處理裝置，其中決定單元包括一重分配器，其從一第一應用程式框架接收輸入資料並供應輸入資料給複數個應用程式框架中之選擇性的一個。
- (8) 如第(5)項所述之資訊處理裝置，其中決定單元包括一第一應用程式框架，其接收輸入資料並選擇性地供應輸入資料給一第一應用程式和複數個應用程式框架中之選擇性的一個之一者。
- (9) 如第(5)項所述之資訊處理裝置，其中決定單元更配置以供應輸入資料給專用於一照相機功能中之一第一應用程式框架和專用於一通訊功

能中之一第二應用程式框架的選擇性之一個。

- (10) 如第(9)項所述之資訊處理裝置，其中在第一應用程式框架上執行之一成像應用程式產生的一影像資料被供應給在第二應用程式框架上執行之一通訊應用程式來透過一網路傳輸給一目標裝置。
- (11) 如第(2)項所述之資訊處理裝置，其中決定單元係配置以基於來自一目前啟動的應用程式之指令來決定複數個應用程式框架之哪個應用程式框架為選擇的應用程式框架。
- (12) 如第(11)項所述之資訊處理裝置，其中決定單元基於決定來控制切換至複數個應用程式框架中之選擇性的一個，並切換至操作在該複數個應用程式框架中之選擇性的一個上的一應用程式。
- (13) 如第(11)項所述之資訊處理裝置，其中決定單元更配置以命令一目前應用程式框架停止操作，並命令複數個應用程式框架中之選擇性的一個開始操作。
- (14) 如第(5)項所述之資訊處理裝置，其中當決定單元從一第一應用程式接收指令以從一第一應用程式框架切換至複數個應用程式框架中之選擇性的一個時，決定單元切換來自第一應用程式框架之輸入資料的一連線目的至複數個應

用程式框架中之選擇性的一個。

(15) 如第(2)項所述之資訊處理裝置，其中一所欲應用程式操作在複數個應用程式框架中之選擇性的一個，且當所欲應用程式被一使用者選擇時，決定單元自動地切換一主動應用程式框架成複數個應用程式框架中之選擇性的一個。

(16) 如第(2)項所述之資訊處理裝置，更包括：
一記憶體，配置以儲存被一第一應用程式框架和可在第一應用程式框架上執行的一第一應用程式之至少一者、和複數個應用程式框架中之選擇性的一個和可在複數個應用程式框架中之選擇性的一個上執行之一第二應用程式之至少一者使用的資料。

(17) 一種資訊處理方法，包括：
執行關聯於一第一應用程式框架的一第一應用程式；
切換一第二應用程式框架的一操作狀態以變成主動；及
在切換第二應用程式框架的操作狀態以變成主動之後執行關聯於第二應用程式框架的一第二應用程式。

(18) 如第(17)項所述之資訊處理方法，更包括：
在切換第二應用程式框架的操作狀態以變成主動之後，透過第二應用程式框架供應一輸入資

料給第二應用程式。

- (19) 如第(17)項所述之資訊處理方法，更包括：
當切換第二應用程式框架的操作狀態以變成主動時，停止第一應用程式的操作。
- (20) 如第(17)項所述之資訊處理方法，其中基於來自第二應用程式的指令來切換第二應用程式框架的操作狀態以變成主動。
- (21) 一種以一程式實作的非暫態電腦可讀媒體，當程式被一電腦執行時使電腦進行一方法，方法包括：
執行關聯於一第一應用程式框架的一第一應用程式；
切換一第二應用程式框架的一操作狀態以變成主動；及
在切換第二應用程式框架的操作狀態以變成主動之後執行關聯於第二應用程式框架的一第二應用程式。
- (22) 一種資訊處理裝置，包括複數個應用程式框架及一切換單元，切換應用程式框架。
- (23) 如第(22)項所述之資訊處理裝置，更包括複數個應用程式框架中的一第一應用程式框架、操作在第一應用程式框架上的一第一應用程式、不同於第一應用程式框架的一第二應用程式框架、及操作在第二應用程式框架上的一第

二應用程式，且其中當命令從第一應用程式切換為第二應用程式框架時，切換單元將裝置的輸入和輸出目的地從第一應用程式框架切換為第二應用程式框架。

(24) 如第(23)項所述之資訊處理裝置，其中切換單元進行切換以一對一地連接複數個應用程式框架和裝置。

(25) 如第(22)至(24)項之任一項所述之資訊處理裝置，其中切換單元分別具有一個實現停止應用程式中的預定應用程式並啟動另一應用程式之函數的部分、及一個實現將複數個應用程式框架中的預定應用程式與裝置分開並將預定應用程式連接至另一應用程式框架之函數的部分。

(26) 如第(23)至(25)項之任一項所述之資訊處理裝置，其中當命令從第一應用程式切換為第二應用程式框架時，切換單元停止第一應用程式並啟動第二應用程式。

(27) 如第(23)至(26)項之任一項所述之資訊處理裝置，其中第一應用程式有複數個，其中切換單元係呈現為複數個第一應用程式中的一個應用程式，且其中切換單元為從裝置傳送輸入給第二應用程式框架的第一應用程式。

(28) 如第(23)至(27)項之任一項所述之資訊處

理裝置，其中切換單元係呈現為第一應用程式框架的一個功能，且其中包括在第一應用程式框架中的切換單元從裝置傳送輸入給第二應用程式框架。

(29) 如第(23)至(28)項之任一項所述之資訊處理裝置，其中來自裝置的輸入係藉由供應包括在一作業系統中之函數的輸入，從第一應用程式或第一應用程式框架供應給第二應用程式框架。

(30) 如第(29)項所述之資訊處理裝置，其中供應函數的輸入係為 `uinput` 的函數。

(31) 如第(23)至(30)項之任一項所述之資訊處理裝置，其中來自第二應用程式的輸出資料係透過第二應用程式框架儲存在儲存單元中，且其中儲存在儲存單元中的輸出資料被第一應用程式讀取而接著透過第一應用程式框架供應給裝置。

(32) 如第(31)項所述之資訊處理裝置，其中第二應用程式所畫的繪圖資料係透過第二應用程式儲存在儲存單元中，且其中儲存在儲存單元中的繪圖資料被第一應用程式讀取而接著透過第一應用程式框架供應給顯示器。

(33) 如第(23)至(32)項之任一項所述之資訊處理裝置，其中第一應用程式框架係關於照相機

功能的應用程式框架，且其中第二應用程式係關於通訊功能的應用程式框架。

(34) 如第(33)項所述之資訊處理裝置，其中根據來自藉由第一應用程式操作在關於照相機功能之第一應用程式框架上的程序由使用者進行的輸入操作所擷取之影像藉由第二應用程式操作在關於通訊功能之第二應用程式框架上的程序傳送給另一裝置。

(35) 一種資訊處理裝置的資訊處理方法，其中資訊處理裝置包括複數個應用程式框架，資訊處理方法包括切換應用程式框架。

(36) 一種使電腦執行切換應用程式框架之程序的程式，其中電腦控制包括複數個應用程式框架的資訊處理裝置。

本揭露包含關於在 2012 年 6 月 11 日申請於日本專利局之揭露於日本優先權專利申請書第 JP 2012-125920 號之主題的主題，特此藉由引用來合併其全部內容。

【符號說明】

10：資訊處理裝置

21：中央處理單元

22：唯讀記憶體

23：隨機存取記憶體

24：內部匯流排

25：輸入和輸出介面
26：輸入單元
27：輸出單元
28：儲存單元
29：通訊單元
30：驅動程式
31：可移除媒體
11-1-11-M：裝置
12-1-12-M：裝置驅動程式
13：作業系統
14：應用程式框架
15-1-15-N：應用程式
100：仲裁器
101：仲裁器
102：仲裁器核心
103：程序切換
104：裝置伺服器
111：圖形伺服器
112：輸入伺服器
113：聲音伺服器
121：應用程式框架
122：應用程式
122-1-122-N：應用程式
131：應用程式框架

- 132：應用程式
- 132-1-132-N：應用程式
- 141：應用程式框架
- 142：應用程式
- 142-1-142-N：應用程式
- 151：圖形裝置介面
- 152：輸入裝置介面
- 161：圖形裝置驅動程式
- 162：輸入裝置驅動程式
- 171：顯示器
- 172：鍵盤
- 173：滑鼠
- 301：應用程式框架
- 302：輸入和輸出決定單元
- 401：共享記憶體

申請專利範圍

1.一種資訊處理裝置，包含：

複數個應用程式框架，在該複數個應用程式框架上執行應用程式；及

一決定單元，配置以控制該複數個應用程式框架之可操作狀態的切換。

2.如申請專利範圍第 1 項所述之資訊處理裝置，其中該決定單元基於來自在該複數個應用程式框架中之選擇性的一個上執行之一應用程式的指令來控制該複數個應用程式框架之可操作狀態的切換。

3.如申請專利範圍第 2 項所述之資訊處理裝置，更包含：

一程序切換單元，配置以切換一主動應用程式為在該複數個應用程式框架中之選擇性的一個上執行之該應用程式。

4.如申請專利範圍第 3 項所述之資訊處理裝置，其中該程序切換單元更配置以停止一目前啟動的應用程式之操作並開始在該複數個應用程式框架中之選擇性的一個上執行之該應用程式的操作。

5.如申請專利範圍第 1 項所述之資訊處理裝置，其中該決定單元更配置以供應一輸入資料給該複數個應用程式框架中之選擇性的一個。

6.如申請專利範圍第 5 項所述之資訊處理裝置，其中該決定單元更配置以基於來自在該複數個應用程式框架中

之選擇性的一個上執行之一應用程式的指令來供應該輸入資料給該複數個應用程式框架中之選擇性的一個。

7.如申請專利範圍第 5 項所述之資訊處理裝置，其中該決定單元包含一重分配器，其從一第一應用程式框架接收該輸入資料並供應該輸入資料給該複數個應用程式框架中之選擇性的一個。

8.如申請專利範圍第 5 項所述之資訊處理裝置，其中該決定單元包含一第一應用程式框架，其接收該輸入資料並選擇性地供應該輸入資料給一第一應用程式和該複數個應用程式框架中之選擇性的一個之一者。

9.如申請專利範圍第 5 項所述之資訊處理裝置，其中該決定單元更配置以供應該輸入資料給專用於一照相機功能中之一第一應用程式框架和專用於一通訊功能中之一第二應用程式框架的選擇性之一個。

10.如申請專利範圍第 9 項所述之資訊處理裝置，其中在該第一應用程式框架上執行之一成像應用程式產生的一影像資料被供應給在該第二應用程式框架上執行之一通訊應用程式來透過一網路傳輸給一目標裝置。

11.如申請專利範圍第 2 項所述之資訊處理裝置，其中該決定單元係配置以基於來自一目前啟動的應用程式之指令來決定該複數個應用程式框架之哪個應用程式框架為選擇的應用程式框架。

12.如申請專利範圍第 11 項所述之資訊處理裝置，其中該決定單元基於該決定來控制切換至該複數個應用程式

框架中之選擇性的一個，並切換至操作在該複數個應用程式框架中之選擇性的一個上的一應用程式。

13.如申請專利範圍第 11 項所述之資訊處理裝置，其中該決定單元更配置以命令一目前應用程式框架停止操作，並命令該複數個應用程式框架中之選擇性的一個開始操作。

14.如申請專利範圍第 5 項所述之資訊處理裝置，其中當該決定單元從一第一應用程式接收指令以從一第一應用程式框架切換至該複數個應用程式框架中之選擇性的一個時，該決定單元切換來自該第一應用程式框架之該輸入資料的一連線目的至該複數個應用程式框架中之選擇性的一個。

15.如申請專利範圍第 2 項所述之資訊處理裝置，其中一所欲應用程式操作在該複數個應用程式框架中之選擇性的一個，且當該所欲應用程式被一使用者選擇時，該決定單元自動地切換一主動應用程式框架成該複數個應用程式框架中之選擇性的一個。

16.如申請專利範圍第 2 項所述之資訊處理裝置，更包含：

一記憶體，配置以儲存被一第一應用程式框架和可在該第一應用程式框架上執行的一第一應用程式之至少一者、和該複數個應用程式框架中之選擇性的一個和可在該複數個應用程式框架中之選擇性的一個上執行之一第二應用程式之至少一者使用的資料。

17.一種資訊處理方法，包含：

執行關聯於一第一應用程式框架的一第一應用程式；

切換一第二應用程式框架的一操作狀態以變成主動；

及

在切換該第二應用程式框架的該操作狀態以變成主動之後執行關聯於該第二應用程式框架的一第二應用程式。

18.如申請專利範圍第 17 項所述之資訊處理方法，更包含：

在切換該第二應用程式框架的該操作狀態以變成主動之後，透過該第二應用程式框架供應一輸入資料給該第二應用程式。

19.如申請專利範圍第 17 項所述之資訊處理方法，更包含：

當切換該第二應用程式框架的該操作狀態以變成主動時，停止該第一應用程式的操作。

20.如申請專利範圍第 17 項所述之資訊處理方法，其中基於來自該第二應用程式的指令來切換該第二應用程式框架的該操作狀態以變成主動。

21.一種以一程式實作的非暫態電腦可讀媒體，當該程式被一電腦執行時使該電腦進行一方法，該方法包含：

執行關聯於一第一應用程式框架的一第一應用程式；

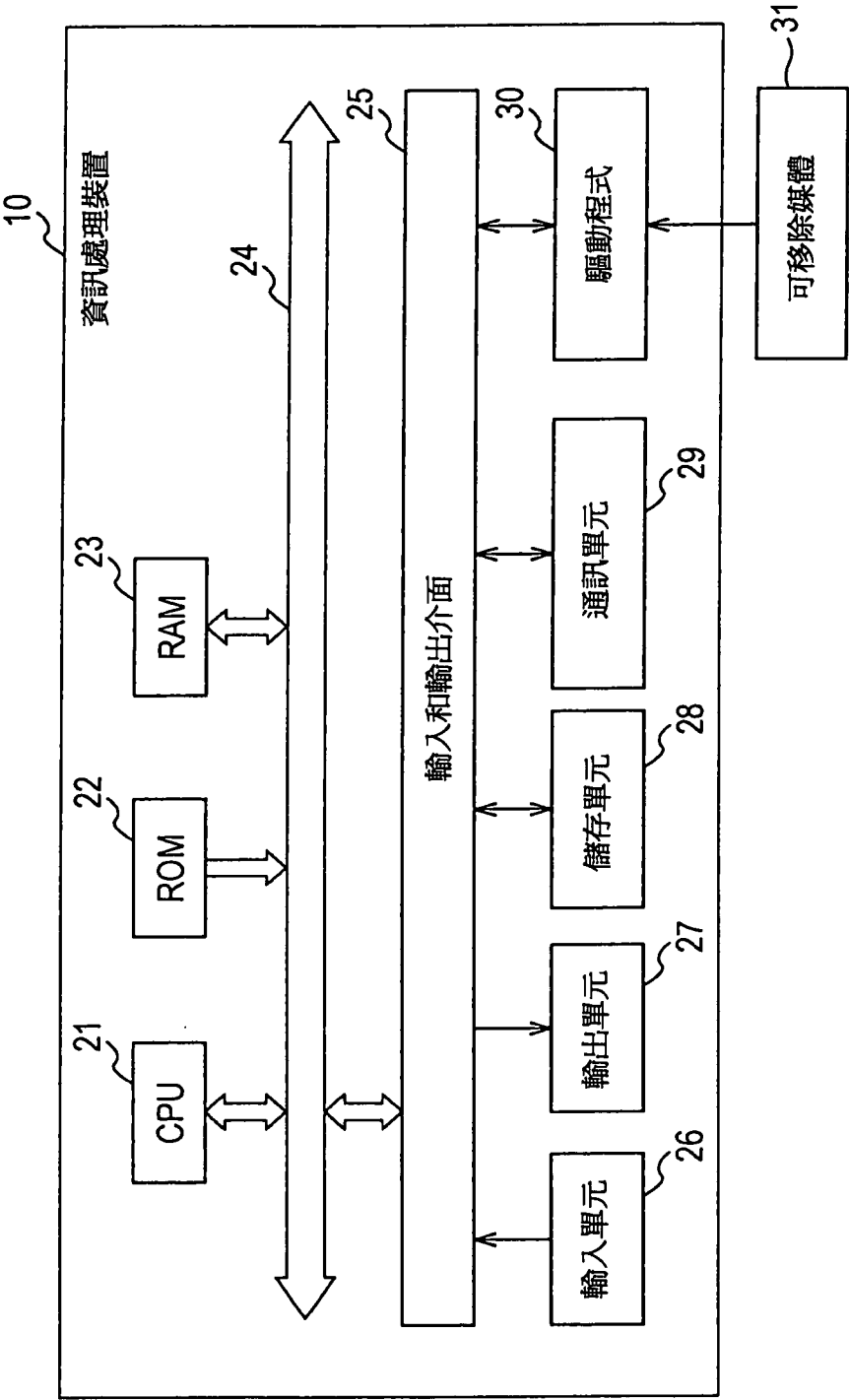
切換一第二應用程式框架的一操作狀態以變成主動；

及

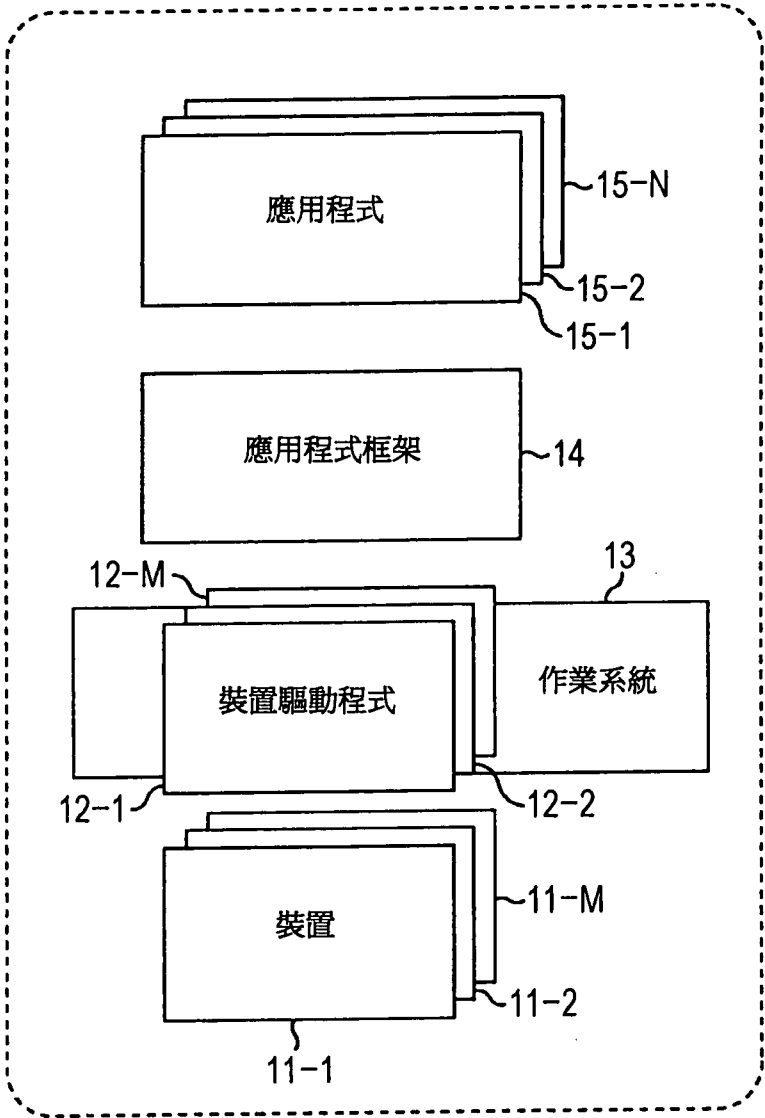
在切換該第二應用程式框架的該操作狀態以變成主動之後執行關聯於該第二應用程式框架的一第二應用程式。

圖 式

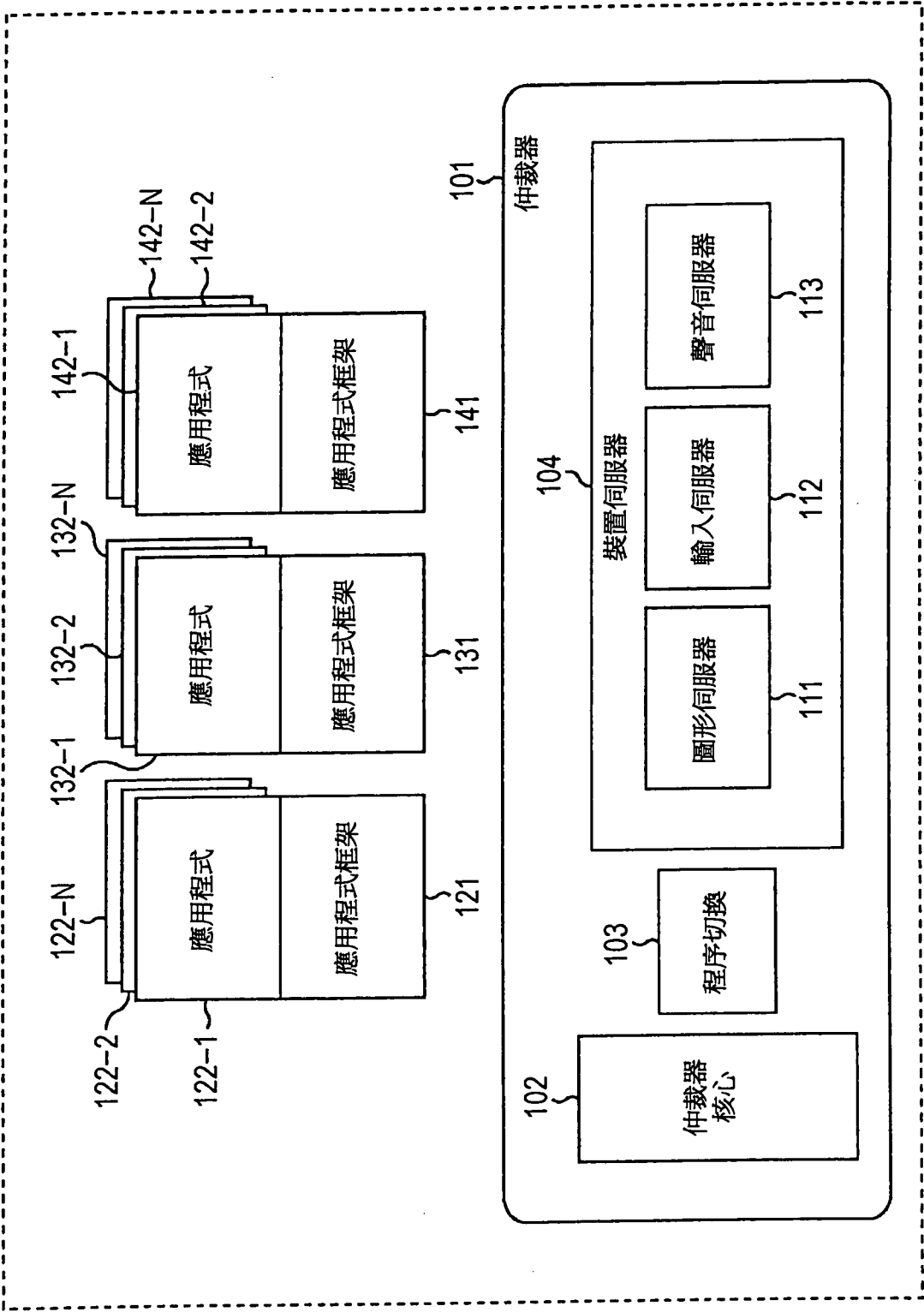
第 1 圖



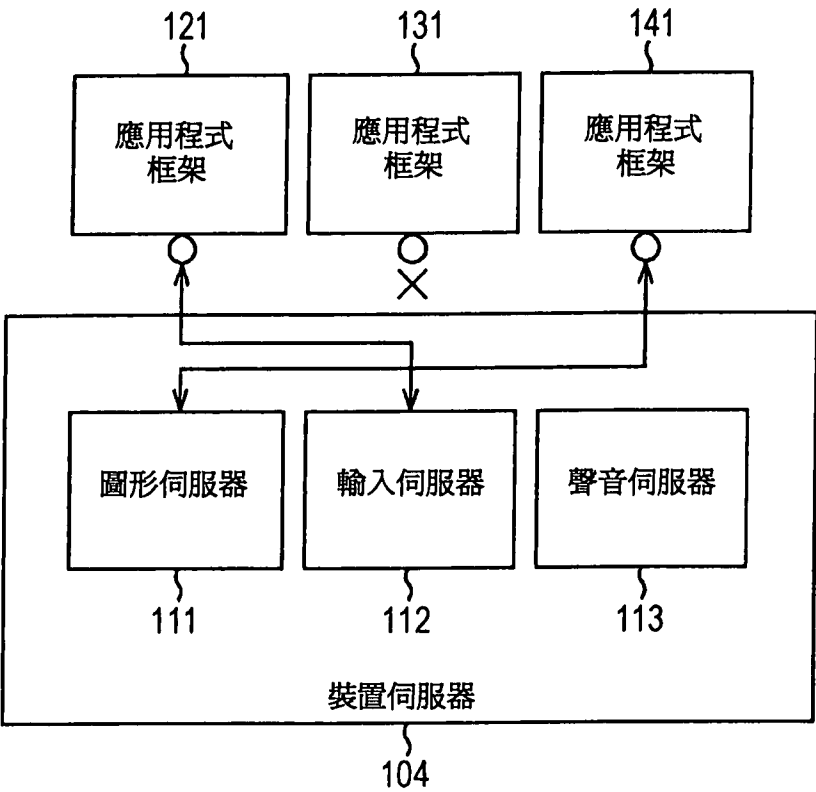
第 2 圖



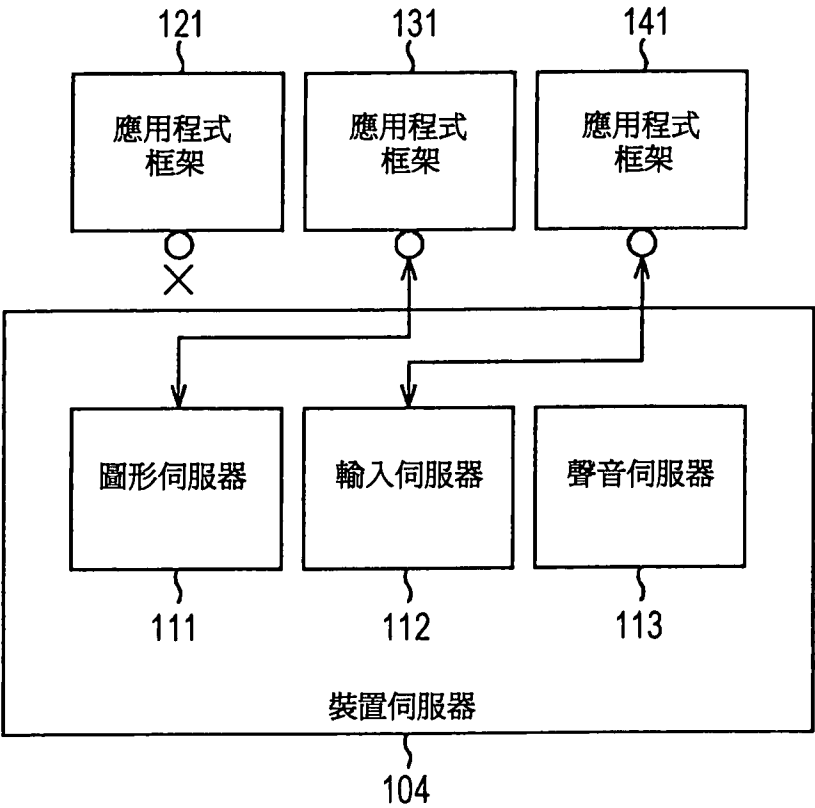
第 3 圖



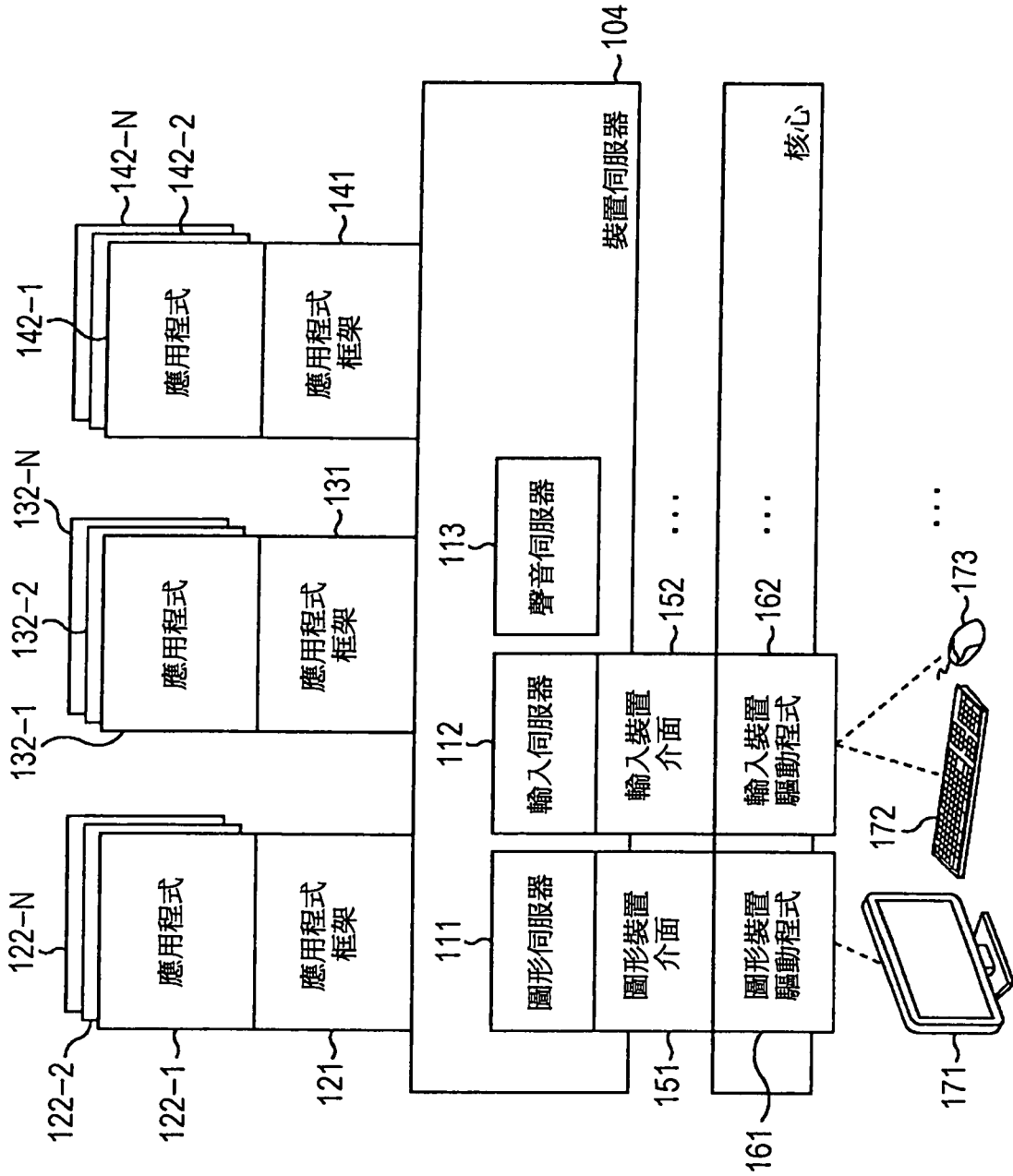
第 4A 圖



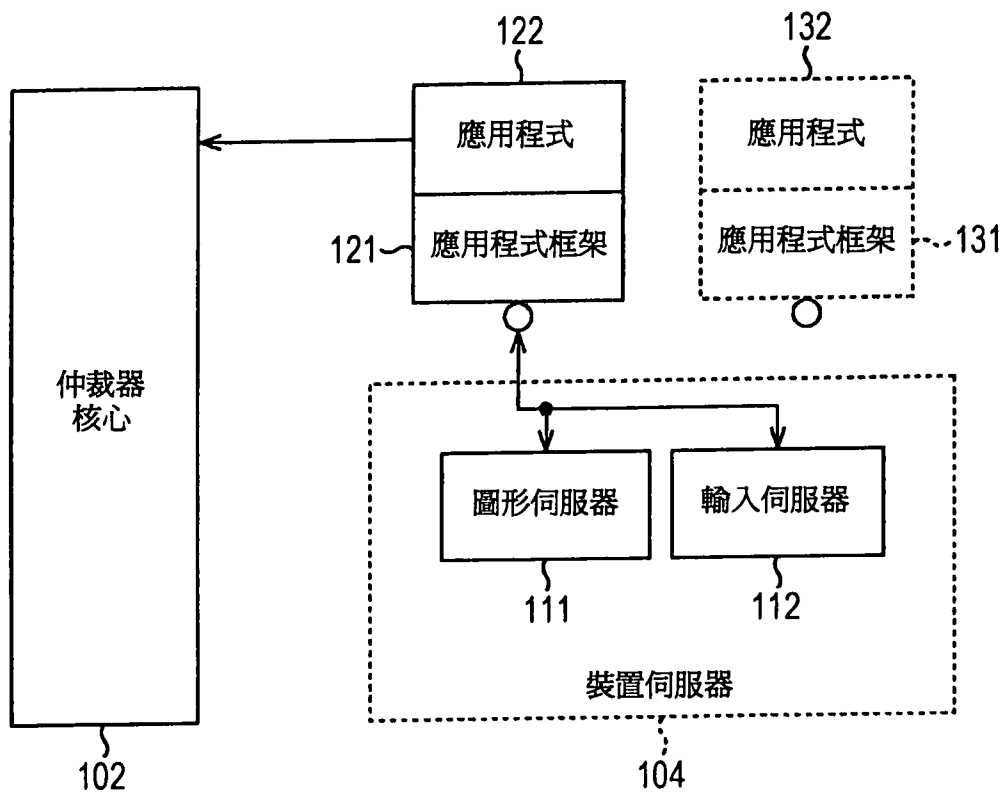
第 4B 圖



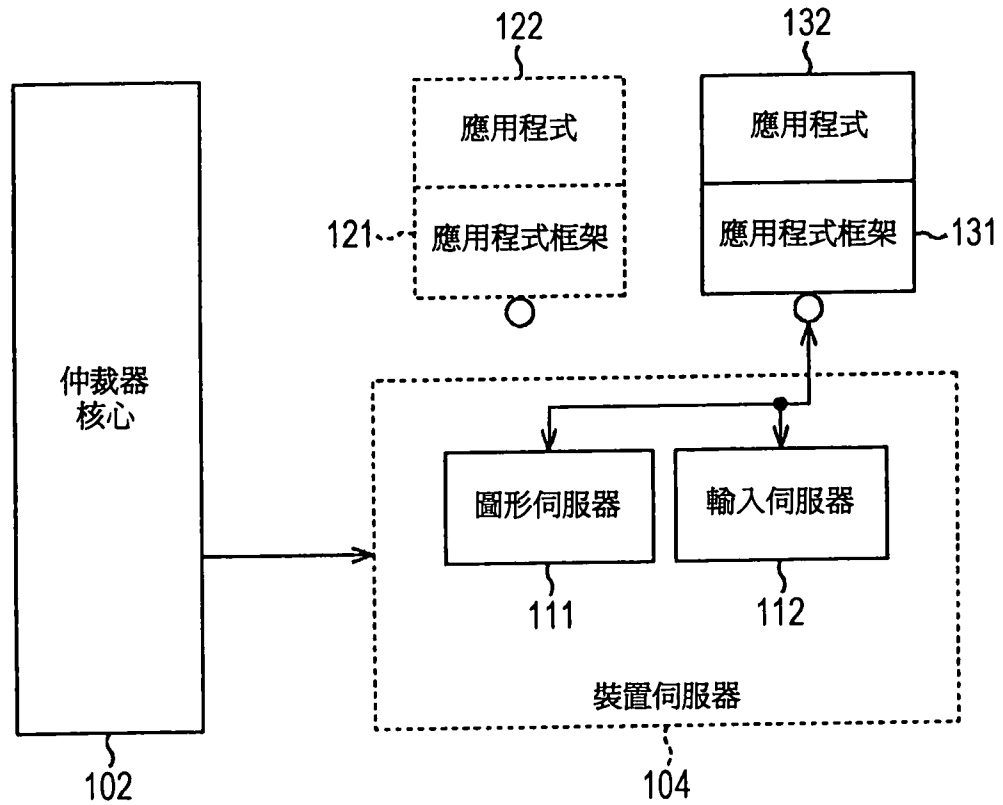
第 5 圖



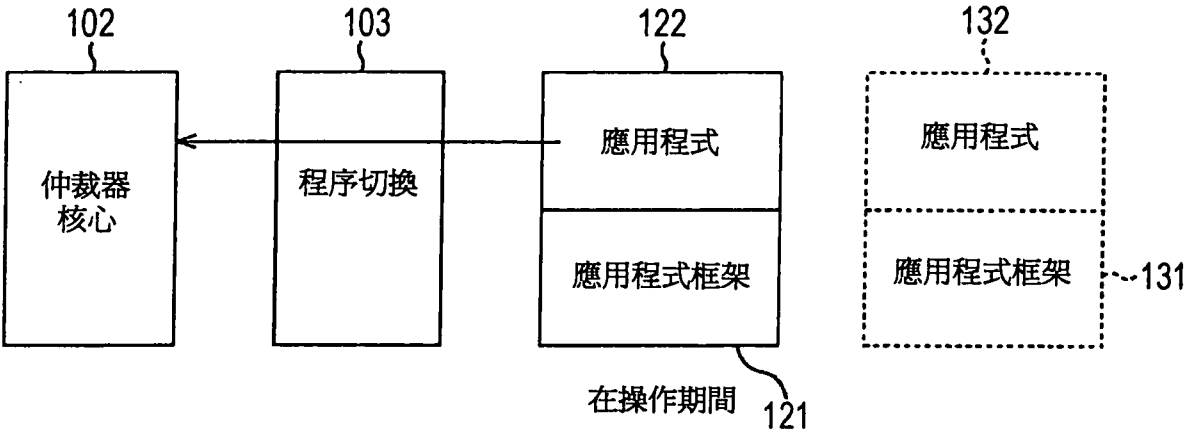
第 6A 圖



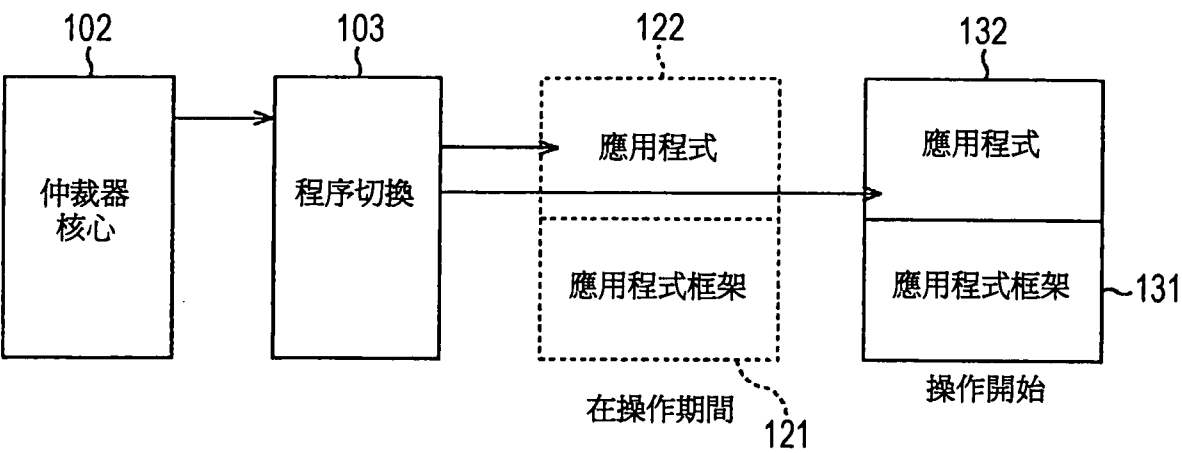
第 6B 圖



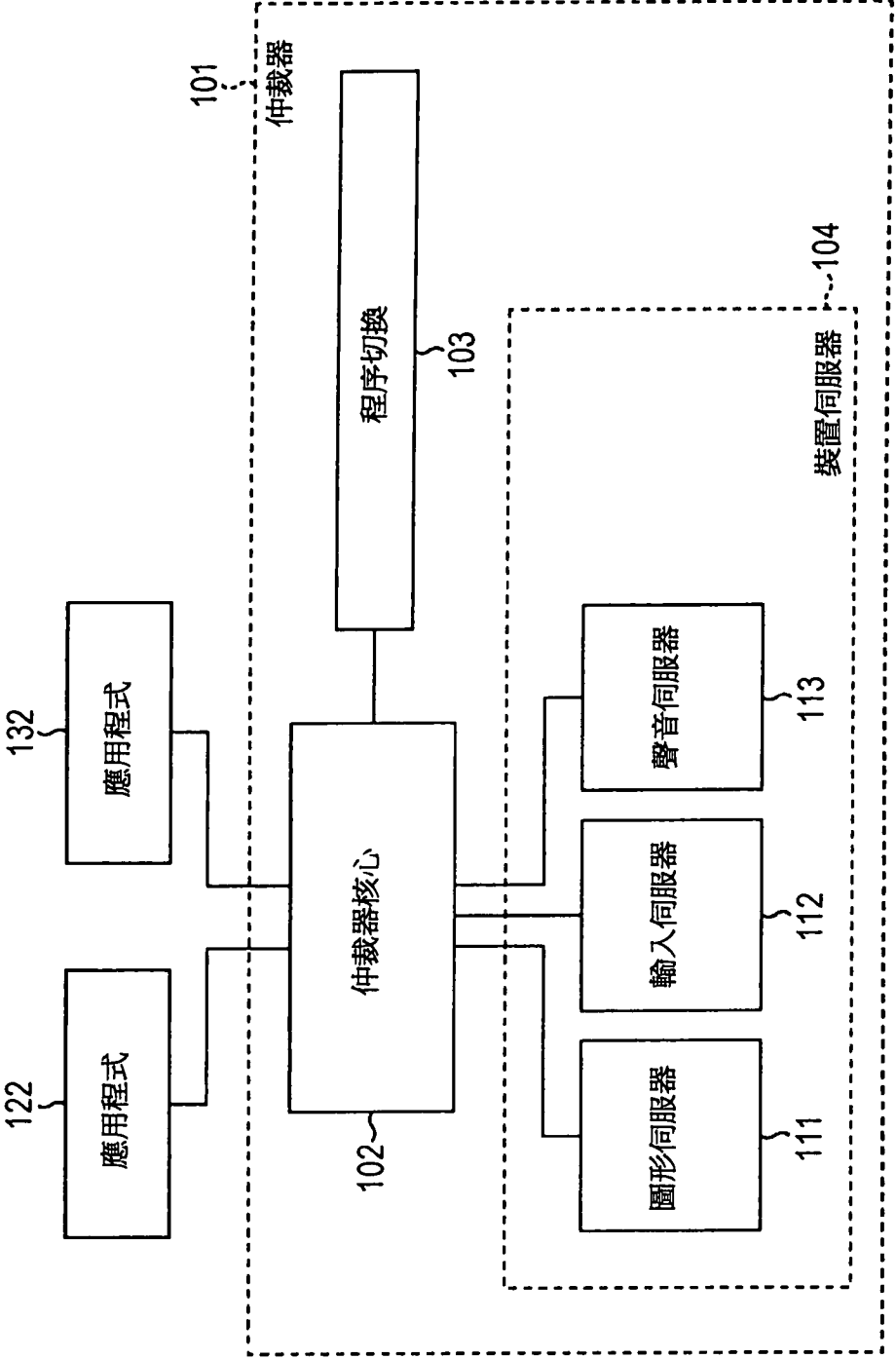
第 7A 圖



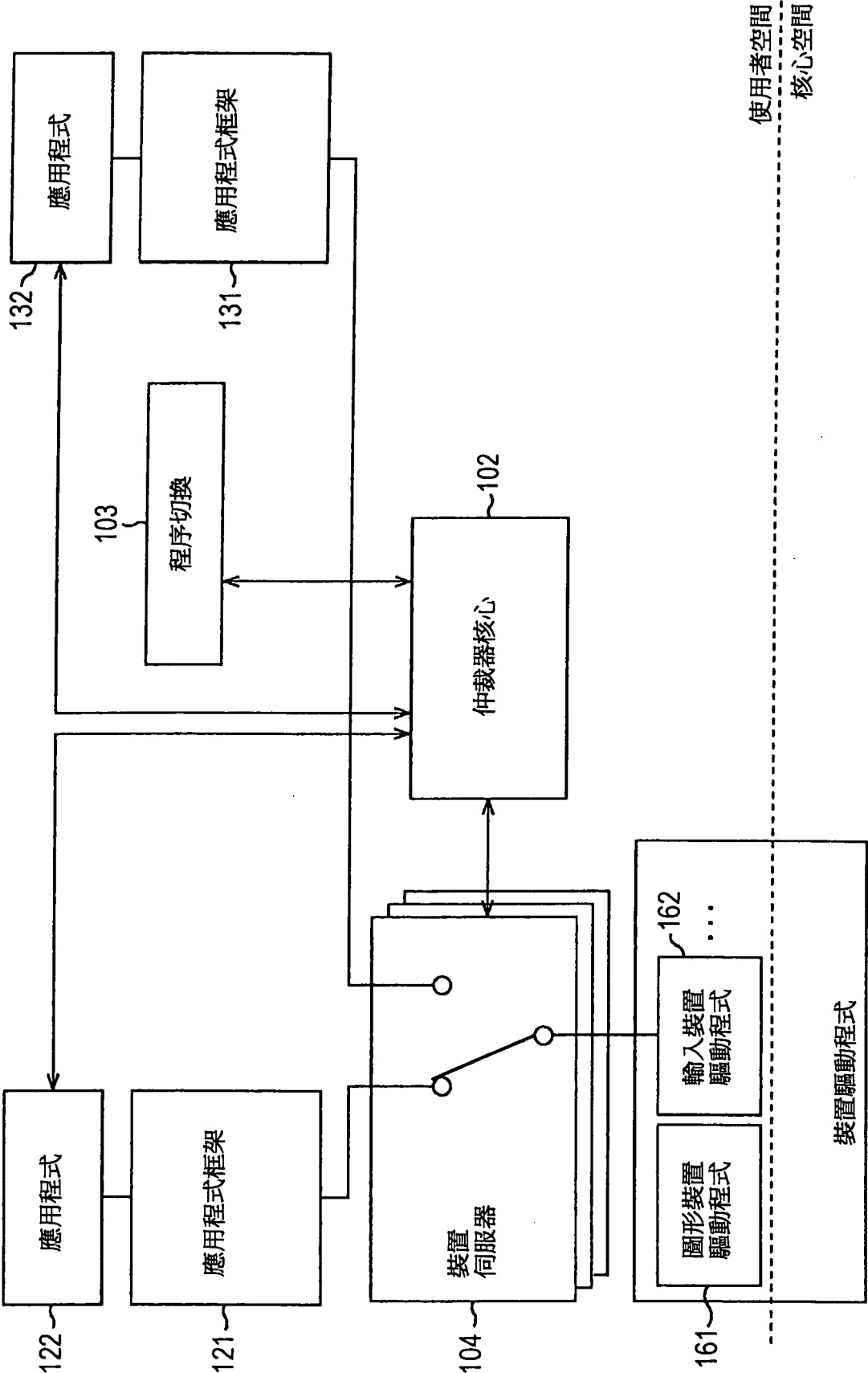
第 7B 圖



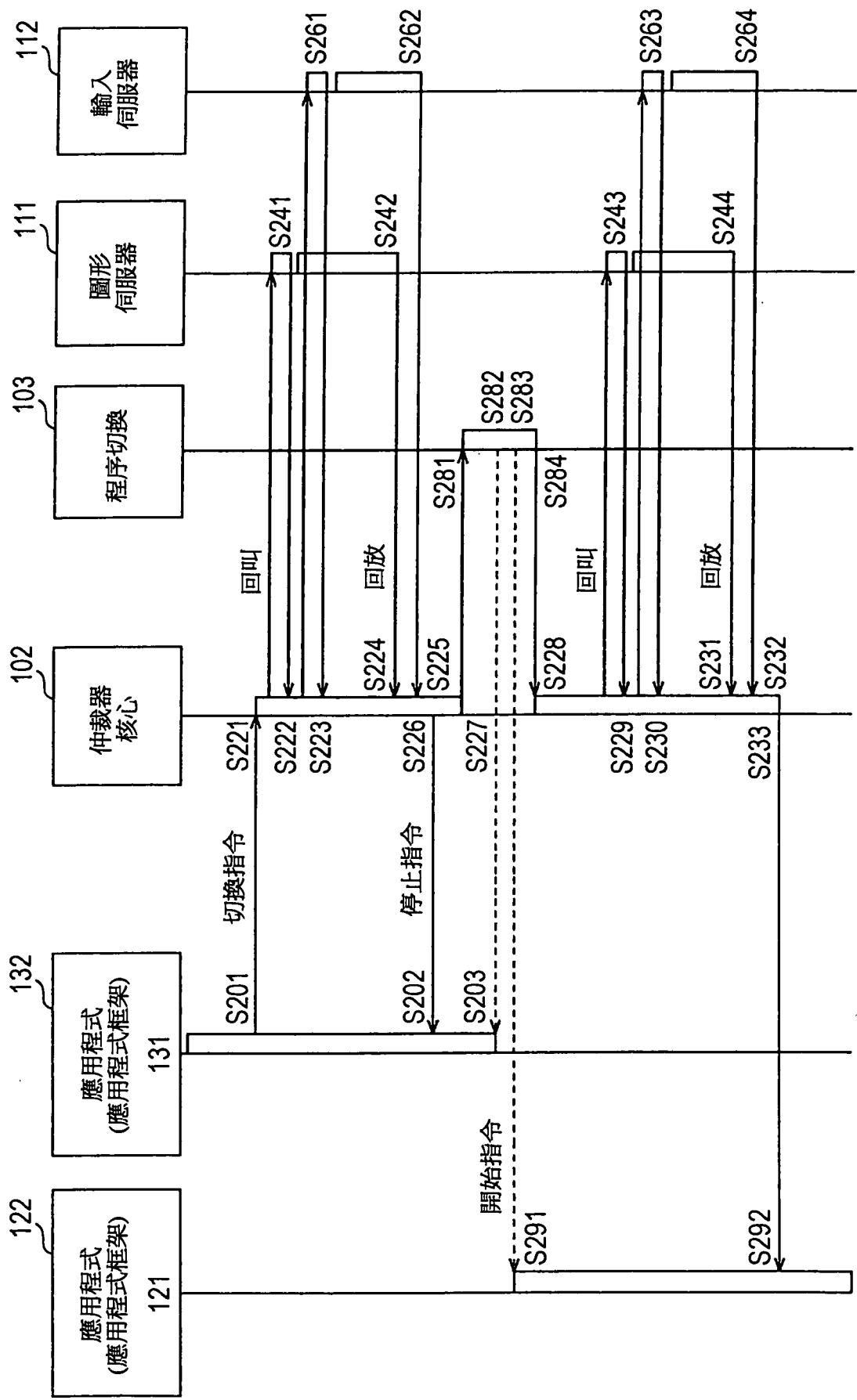
第 8 圖



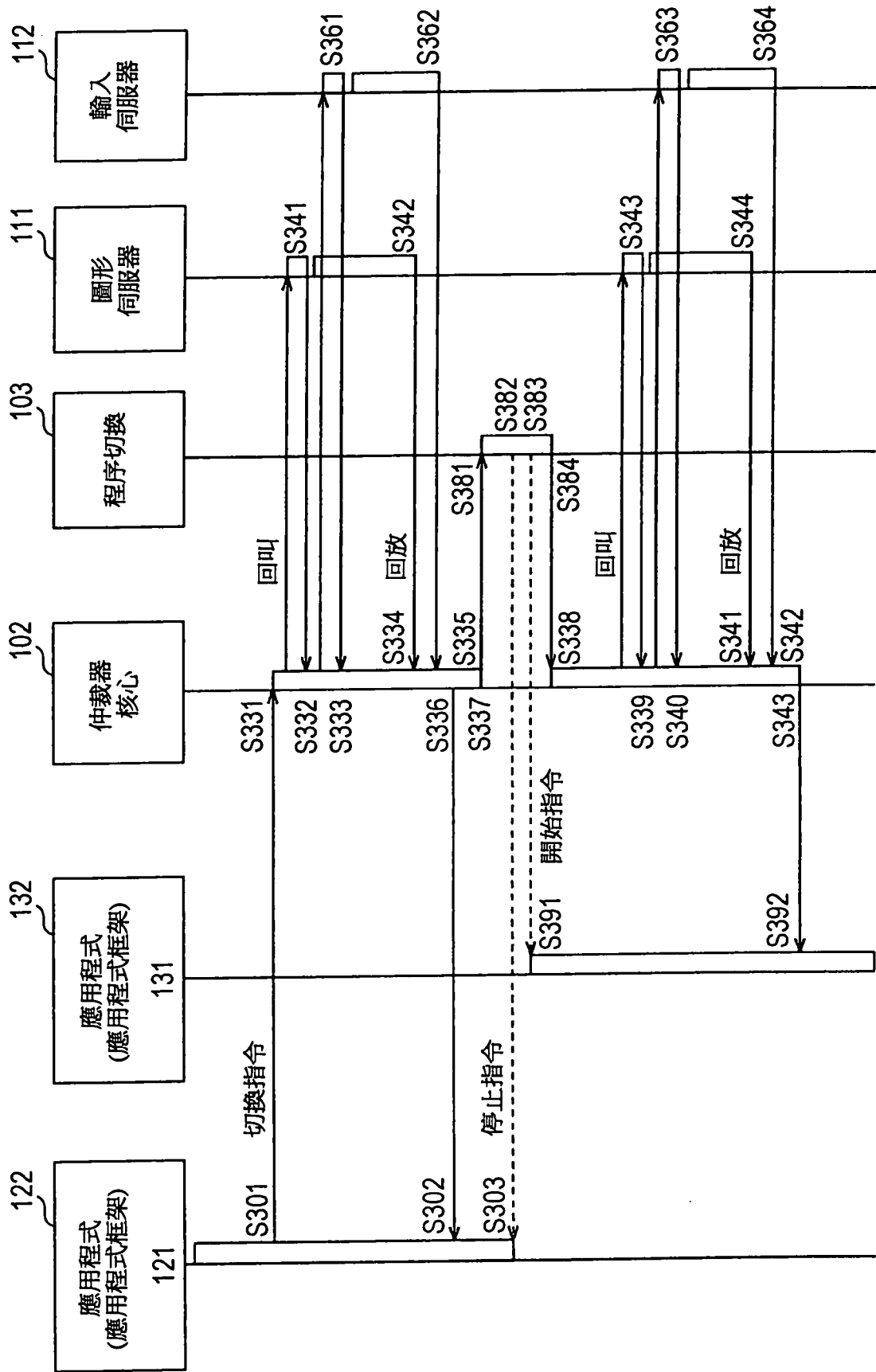
第 9 圖



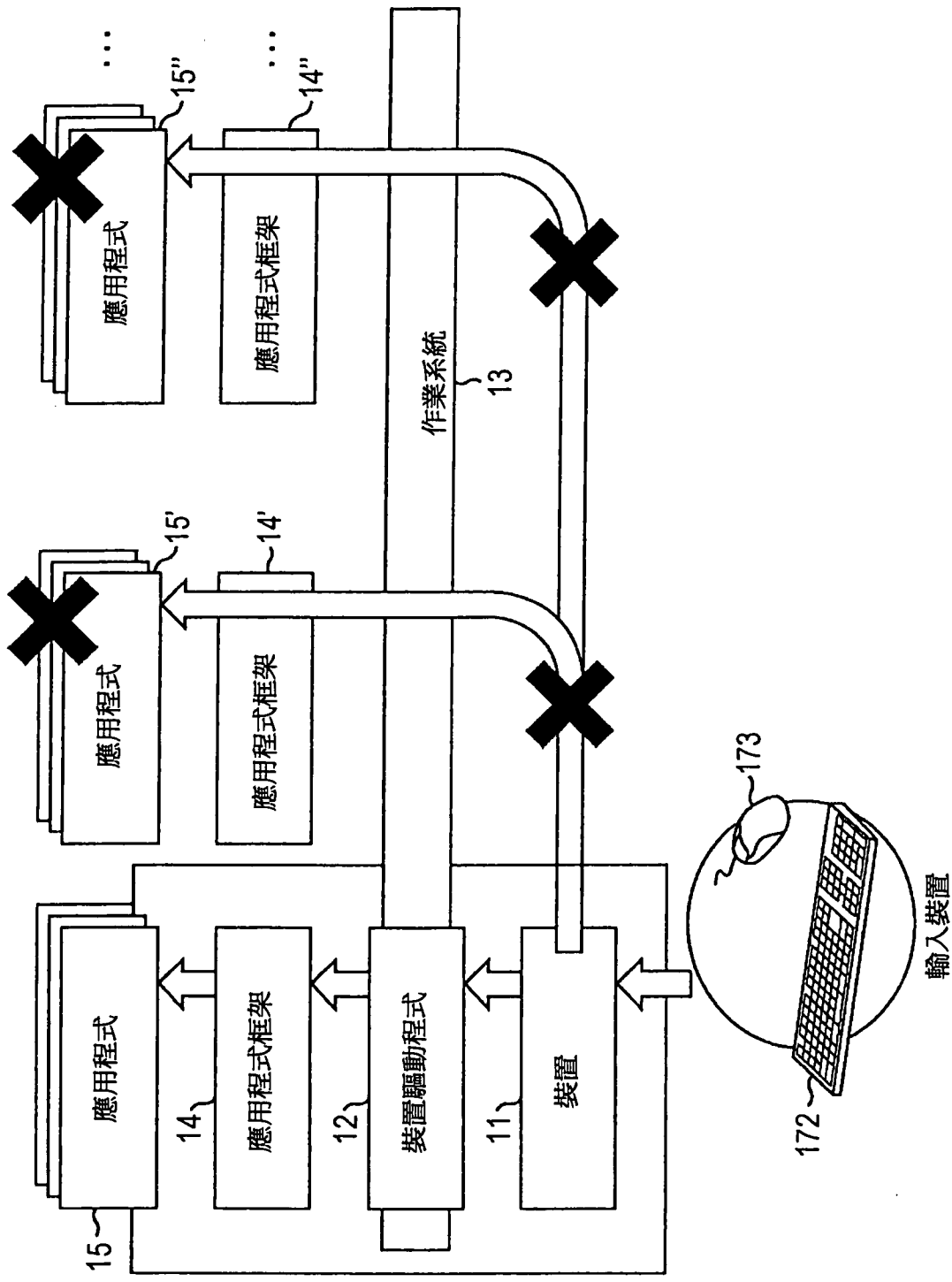
第 11 圖



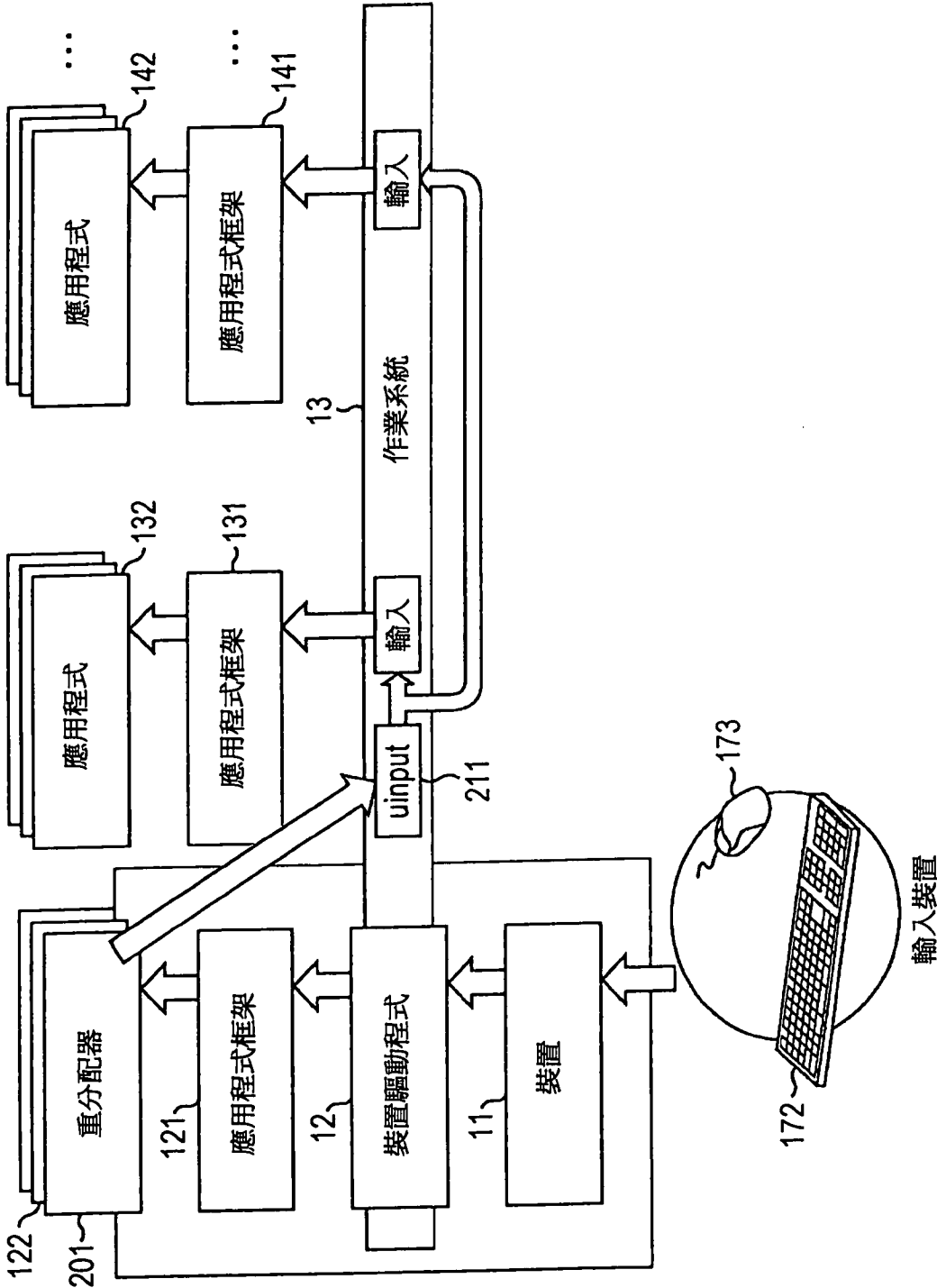
第 12 圖



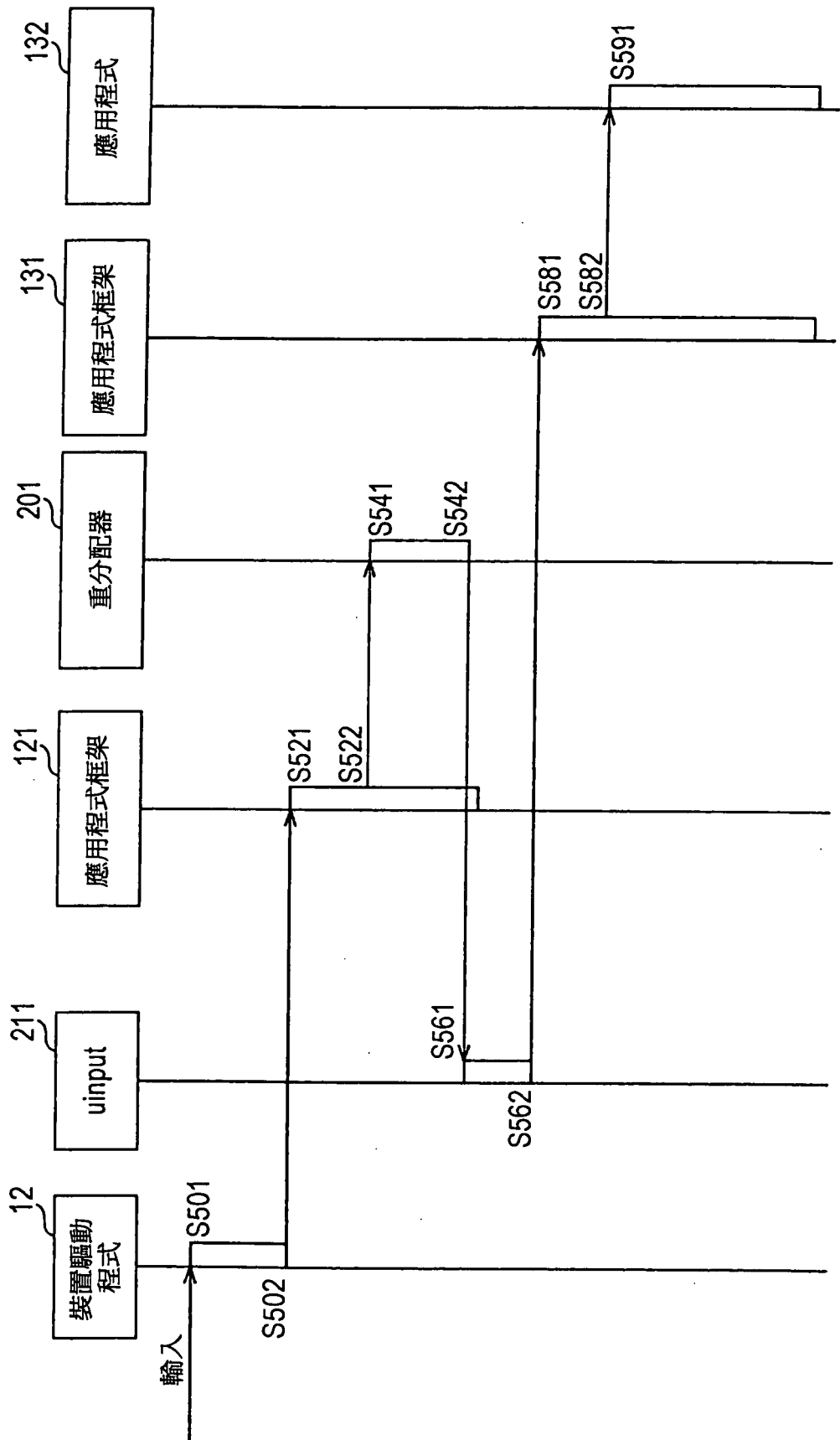
第 13 圖



第 14 圖



第 15 圖



第 16 圖

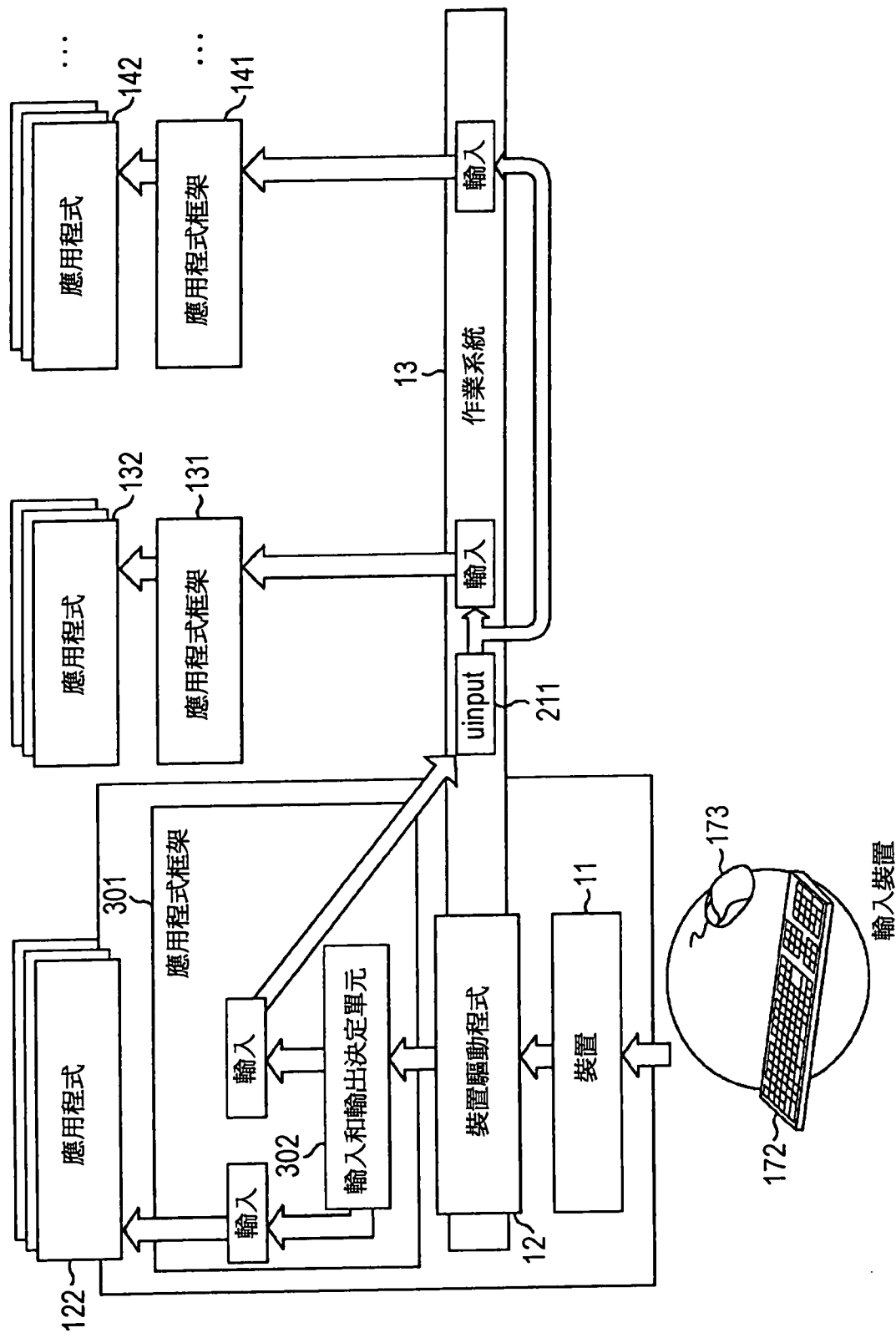
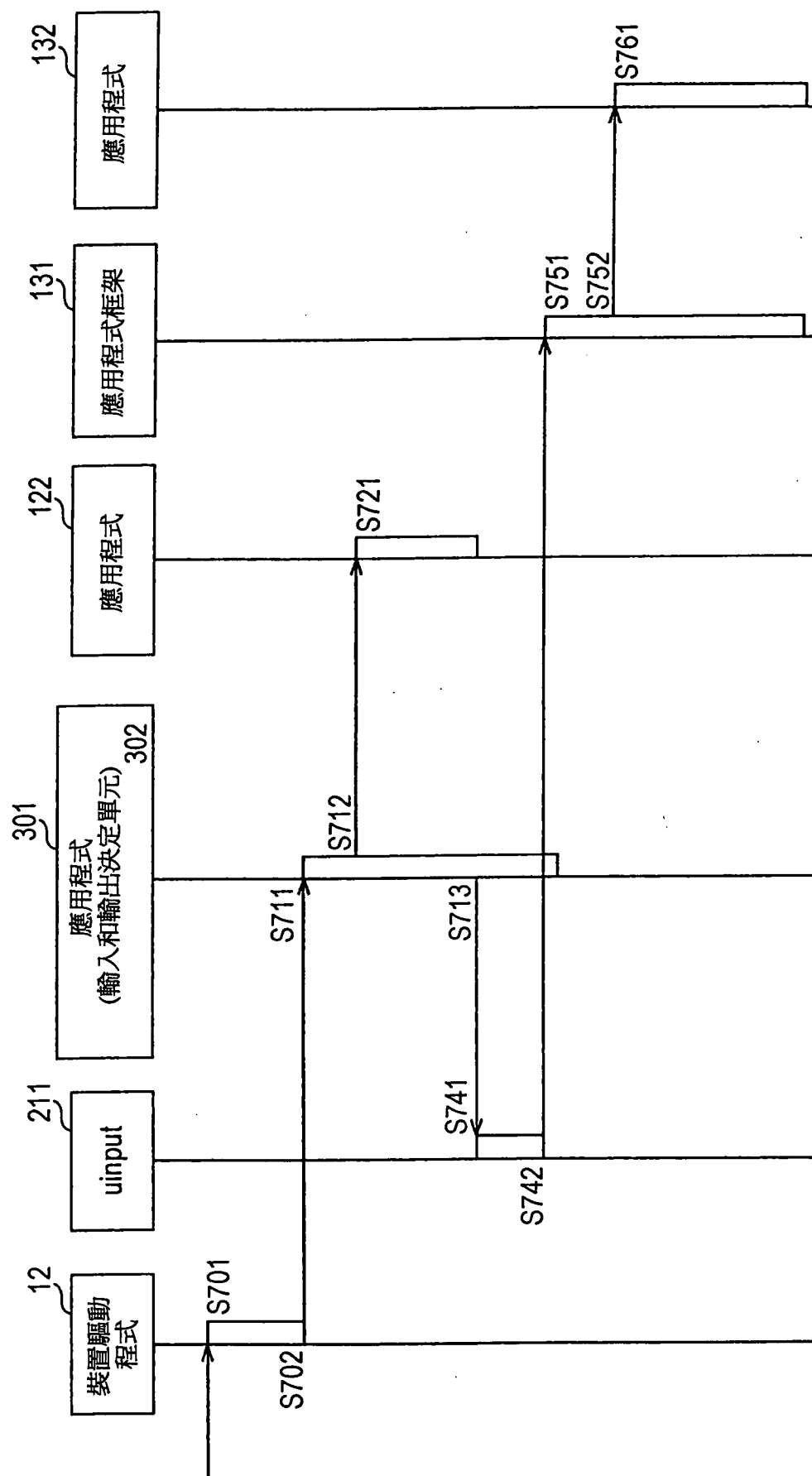
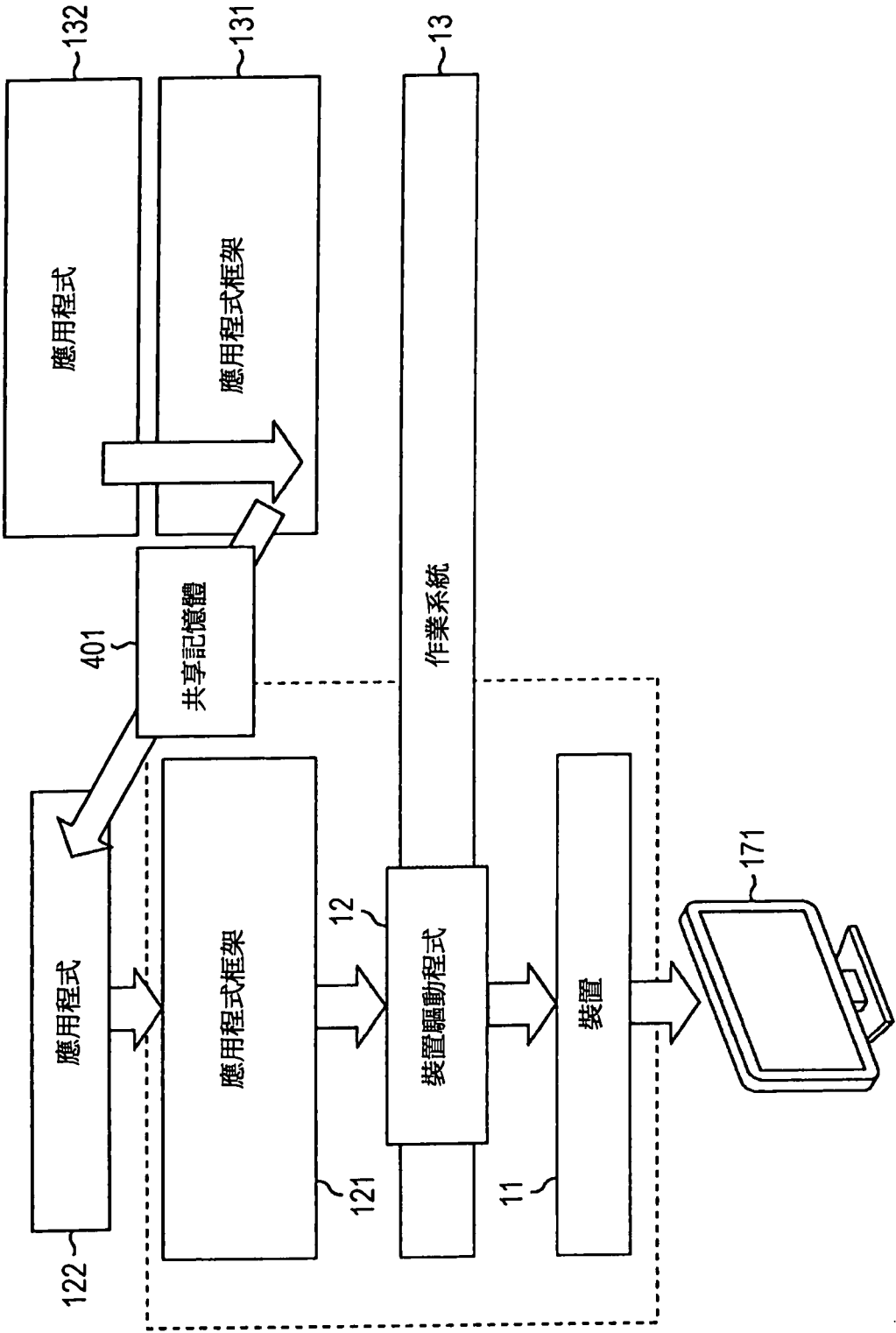


圖
17
錄

第 18 圖



第 19 圖

