



(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201316180 A1

(43)公開日：中華民國 102 (2013) 年 04 月 16 日

(21)申請案號：101129621

(22)申請日：中華民國 101 (2012) 年 08 月 15 日

(51)Int. Cl.：

G06F15/163 (2006.01)

G06F9/50 (2006.01)

(30)優先權：2011/09/01

美國

13/224,198

(71)申請人：高通公司 (美國) QUALCOMM INCORPORATED (US)

美國

(72)發明人：加爾葛許 諾曼 S GARGASH, NORMAN S. (US)；維傑亞瑞真 維諾得

VIJAYARAJAN, VINOD (IN)

(74)代理人：陳長文

申請實體審查：有 申請專利範圍項數：44 項 圖式數：11 共 79 頁

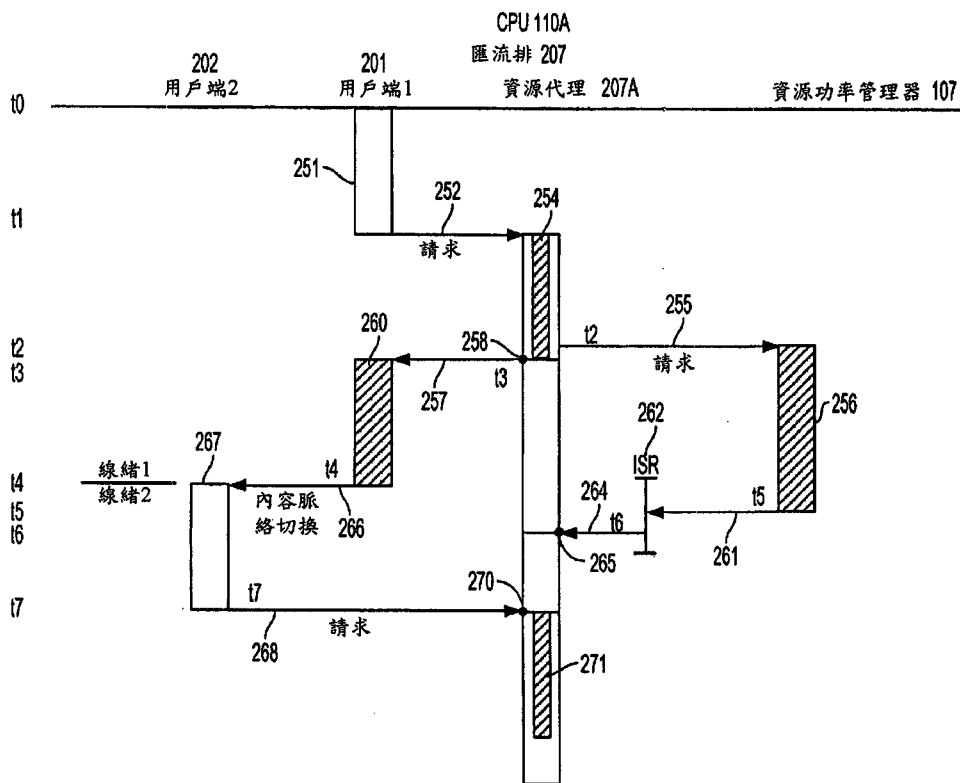
(54)名稱

用於管理可攜式計算裝置上之平行資源請求之方法及系統

METHOD AND SYSTEM FOR MANAGING PARALLEL RESOURCE REQUESTS IN A PORTABLE COMPUTING DEVICE

(57)摘要

本發明描述一種用於管理可攜式計算裝置(「PCD」)中之平行資源請求之方法及系統。該系統及方法包括自第一用戶端產生第一請求，該第一請求係在第一執行線緒之內容脈絡中發出的。可將該第一請求轉遞至一資源。該資源可應答該第一請求並起始非同步處理。該資源可處理該第一請求同時允許第一用戶端繼續第一執行線緒中之處理。該資源可傳信第一請求之處理的完成且可接收第二請求。該第二請求導致第一請求之處理的完成。第一請求之處理的完成可包括將資源之本端代表更新至一新狀態並調用任何登記之回叫。該資源可變得可用於服務於第二請求，且可處理第二請求。



107：資源功率管理器
110A：中央處理單元 (CPU)

201：第一用戶端

202：第二用戶端

207：系統匯流排

207A：資源代理

251：參考數字

252：參考數字

254：參考數字

255：參考數字

256：參考數字

257：參考數字

258：點

260：參考數字

261：參考數字

264：參考數字

265：點

266：參考數字

268：參考數字

270：點

271：時間週期



(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201316180 A1

(43)公開日：中華民國 102 (2013) 年 04 月 16 日

(21)申請案號：101129621

(22)申請日：中華民國 101 (2012) 年 08 月 15 日

(51)Int. Cl. : **G06F15/163 (2006.01)**

G06F9/50 (2006.01)

(30)優先權：2011/09/01 美國

13/224,198

(71)申請人：高通公司 (美國) QUALCOMM INCORPORATED (US)

美國

(72)發明人：加爾葛許 諾曼 S GARGASH, NORMAN S. (US) ; 維傑亞瑞真 維諾得

VIJAYARAJAN, VINOD (IN)

(74)代理人：陳長文

申請實體審查：有 申請專利範圍項數：44 項 圖式數：11 共 79 頁

(54)名稱

用於管理可攜式計算裝置上之平行資源請求之方法及系統

METHOD AND SYSTEM FOR MANAGING PARALLEL RESOURCE REQUESTS IN A PORTABLE COMPUTING DEVICE

(57)摘要

本發明描述一種用於管理可攜式計算裝置(「PCD」)中之平行資源請求之方法及系統。該系統及方法包括自第一用戶端產生第一請求，該第一請求係在第一執行線緒之內容脈絡中發出的。可將該第一請求轉遞至一資源。該資源可應答該第一請求並起始非同步處理。該資源可處理該第一請求同時允許第一用戶端繼續第一執行線緒中之處理。該資源可傳信第一請求之處理的完成且可接收第二請求。該第二請求導致第一請求之處理的完成。第一請求之處理的完成可包括將資源之本端代表更新至一新狀態並調用任何登記之回叫。該資源可變得可用於服務於第二請求，且可處理第二請求。

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：101129621

※ 申請日：101.8.15

※IPC 分類：G06F 15/163 (2006.01)

一、發明名稱：(中文/英文)

G06F 9/50 (2006.01)

用於管理可攜式計算裝置上之平行資源請求之方法及系統

METHOD AND SYSTEM FOR MANAGING PARALLEL RESOURCE
REQUESTS IN A PORTABLE COMPUTING DEVICE

二、中文發明摘要：

本發明描述一種用於管理可攜式計算裝置(「PCD」)中之平行資源請求之方法及系統。該系統及方法包括自第一用戶端產生第一請求，該第一請求係在第一執行線緒之內容脈絡中發出的。可將該第一請求轉遞至一資源。該資源可應答該第一請求並起始非同步處理。該資源可處理該第一請求同時允許第一用戶端繼續第一執行線緒中之處理。該資源可傳信第一請求之處理的完成且可接收第二請求。該第二請求導致第一請求之處理的完成。第一請求之處理的完成可包括將資源之本端代表更新至一新狀態並調用任何登記之回叫。該資源可變得可用於服務於第二請求，且可處理第二請求。

三、英文發明摘要：

A method and system for managing parallel resource requests in a portable computing device ("PCD") are described. The system and method includes generating a first request from a first client, the first request issued in the context of a first execution thread. The first request may be forwarded to a resource. The resource may acknowledge the first request and initiate asynchronous processing. The resource may process the first request while allowing the first client to continue processing in the first execution thread. The resource may signal completion of the processing of the first request and may receive a second request. The second request causes completion of the processing of the first request. The completion of the processing of the first request may include updating a local representation of the resource to a new state and invoking any registered callbacks. The resource may become available to service the second request, and may process the second request.

四、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：

107	資源功率管理器
110A	中央處理單元(CPU)
201	第一用戶端
202	第二用戶端
207	系統匯流排
207A	資源代理
251	參考數字
252	參考數字
254	參考數字
255	參考數字
256	參考數字
257	參考數字
258	點
260	參考數字
261	參考數字
264	參考數字
265	點
266	參考數字
268	參考數字
270	點
271	時間週期

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

(無)

六、發明說明：

【先前技術】

可攜式計算裝置(「PCD」)正變得日益風行。此等裝置可包括蜂巢式電話、可攜式/個人數位助理(「PDA」)、可攜式遊戲機、可攜式導航單元、掌上型電腦及其他可攜式電子裝置。

PCD可執行各種類型之軟體以用於提供各種功能及特徵。舉例而言，PCD可執行娛樂軟體，其可提供諸如觀看視訊及玩視訊遊戲的功能。PCD亦可支援其他類型之軟體，諸如商業軟體或書寫軟體(諸如試算表、電子郵件及/或文書處理軟體)。

通常，上文所描述之執行於PCD上之軟體需要來自被連結在一起之各種硬體元件的動作。軟體與硬體元件之間的互動可由總操作構架來控制，該總操作構架可被視為一經連結節點結構。在一些個例中，此等元件之間的互動係同步發生，其中針對一特定資源之請求暫時中止一元件之操作直至該請求被應答並准予為止。在其他個例中，此等元件之間的互動係非同步發生，其中當處理針對一特定資源之請求時該請求不暫時中止元件之操作。

然而，將需要使請求元件能夠判定當處理該請求時是否允許資源繼續操作。

【發明內容】

本發明描述一種用於管理可攜式計算裝置中之平行資源請求之方法及系統。在一實施例中，該方法及系統包括自

第一用戶端產生第一請求，該第一請求係在第一執行線緒之內容脈絡中發出的。可將該第一請求轉遞至一資源。該資源可應答該第一請求並起始非同步處理。該資源可處理該第一請求同時允許第一用戶端繼續第一執行線緒中之處理。該資源可傳信第一請求之處理的完成且可接收第二請求。該第二請求導致第一請求之處理的完成。第一請求之處理的完成可包括將資源之本端代表更新至一新狀態並調用任何登記之回叫。該資源可變得可用於服務於第二請求，且可處理第二請求。

【實施方式】

在諸圖中，除非另有指示，否則貫穿各視圖，同樣的參考數字指代同樣的部分。對於諸如「102A」或「102B」之具有字母字元名稱的參考數字而言，該等字母字元名稱可區分存在於同一圖中之兩個同樣的部分或元件。當參考數字意欲涵蓋在所有圖中具有相同參考數字之所有部分時，可省略參考數字之字母字元名稱。

詞語「例示性」在本文中用以意謂「充當一實例、個例或說明」。不必將本文中描述為「例示性」之任何態樣解釋為較佳或優於其他態樣。

在此描述中，術語「應用程式」亦可包括具有可執行內容之檔案，諸如：目的碼、指令碼、位元組碼、標記語言檔案及修補程式。另外，本文中所引用之「應用程式」亦可包括實質上不可執行之檔案(諸如，可能需要開啟之文件或需要存取之其他資料檔案)。

術語「內容」亦可包括具有可執行內容之檔案，諸如：目的碼、指令碼、位元組碼、標記語言檔案及修補程式。另外，本文中所提及之「內容」亦可包括實質上不可執行之檔案(諸如，可能需要開啟之文件或需要存取之其他資料檔案)。

如在此描述中所使用，術語「組件」、「資料庫」、「模組」、「系統」及其類似者意欲指代電腦相關實體，其可為硬體、韌體、硬體與軟體之組合、軟體或執行中之軟體。舉例而言，組件可為(但不限於)在處理器上執行之程序、處理器、物件、可執行程式、執行線緒、程式及/或電腦。藉由說明，執行於計算裝置上之應用程式與該計算裝置兩者皆可為組件。一或多個組件可駐存於程序及/或執行線緒內，且一組件可定位於一電腦上及/或分散於兩個或兩個以上電腦之間。另外，此等組件可自各種電腦可讀媒體執行，該等電腦可讀媒體具有儲存於其上之各種資料結構。該等組件可(諸如)根據具有一或多個資料封包(例如，來自與本端系統、分佈式系統中之另一組件互動及/或藉由信號而跨越諸如網際網路之網路與其他系統互動之組件的資料)之信號藉由本端及/或遠端程序而通信。

在此描述中，術語「通信裝置」、「無線裝置」、「無線電話」、「無線通信裝置」及「無線手機」可互換地使用。隨著第三代(「3G」)及第四代(「4G」)無線技術之出現，更大的頻寬可用性已允許實現更多的具有廣泛多種無線能力的可攜式計算裝置。

在此描述中，術語「可攜式計算裝置」(「PCD」)用以描述依靠有限容量之電源供應器(諸如電池)操作之任何裝置。雖然電池操作之PCD已使用了數十年，但與第三代(「3G」)及第四代(「4G」)無線技術之出現相結合的可再充電電池之技術進步已允許實現具有多種能力的眾多PCD。因此，PCD可尤其為蜂巢式電話、衛星電話、尋呼機、個人數位助理(「PDA」)、智慧電話、導航裝置、智慧型筆電或閱讀器、媒體播放器、上述裝置之組合及具有無線連接之膝上型電腦。

圖1係用於實施用於管理可攜式計算裝置中之平行資源請求之方法及系統之PCD 100(呈無線電話之形式)的例示性、非限制性態樣的功能方塊圖。如所示，PCD 100包括一晶載系統102，該晶載系統102具有一多核心、中央處理單元(「CPU」)110A、一圖形處理器110B及一類比信號處理器126。如熟習此項技術者已知，此等處理器110A、110B、126可在一或多個系統匯流排或另一互連架構上耦接在一起。

如一般熟習此項技術者所理解，CPU 110A可包含第零核心222、第一核心224及第N核心226。在一替代性實施例中，如一般熟習此項技術者所理解，代替使用CPU 110A及圖形處理器110B，亦可使用一或多個數位信號處理器(「DSP」)。另外，亦可使用兩個或兩個以上多核心處理器。

PCD 100可包含由處理器110執行之內部晶片匯流排

(「ICB」)驅動器模組103。一般熟習此項技術者將認識到，在不背離本發明的情況下，每一ICB驅動器模組103可包含一或多個軟體模組，該一或多個軟體模組可劃分為各個部分且由不同處理器110、126執行。

可存在兩種類型之ICB驅動器模組103：上層(「UL」)類型103A；及下層(「LL」)類型103B。大體而言，UL ICB驅動器類型103A將通常由可支援各種應用程式模組105之一或多個處理器110、126執行。LL ICB驅動器類型103B將通常由一個稱作資源功率管理器107之硬體元件執行。

執行LL ICB驅動器103B之資源功率管理器107將大體負責應用及設定頻寬值。此等頻寬值將由資源功率管理器107應用於下文結合圖2所描述之一或多個匯流排及/或開關組構。資源功率管理器107大體負責設定開關組構及匯流排之時脈速度以及從屬件(slave)之時脈速度。從屬件大體上為支援來自執行應用程式105之主控處理器110之請求的硬體組件。

與資源功率管理器107組合之ICB驅動器103A、103B允許針對可存在於類似之開關組構內及/或存在於不同之開關組構中的硬體組件在執行時間動態地產生主從對。ICB驅動器103A、103B及資源功率管理器107可即時地(on-the-fly)或實時地計算及調整用於開關組構及匯流排之頻寬。

在一特定態樣中，本文中所描述之方法步驟中之一或多者可藉由儲存於記憶體112中之包括ICB驅動器103A、103B的可執行指令及參數來實施。形成ICB驅動器103A、

103B之此等指令可由CPU 110A、類比信號處理器126及資源功率管理器107來執行。另外，處理器110A、126、資源功率管理器107、記憶體112、儲存於其中之指令或其組合可充當用於執行本文中所描述之方法步驟中之一或多者的構件。

如圖1中所說明，顯示控制器128及觸控螢幕控制器130耦接至多核心CPU 110A。位於晶載系統102外部之觸控螢幕顯示器132耦接至顯示控制器128及觸控螢幕控制器130。

圖1亦說明一視訊編碼器/解碼器(「編碼解碼器」)134，例如，相交替線(「PAL」)編碼器、電視顯示技術(sequential couleur avec memoire)(「SECAM」)編碼器、全國電視系統委員會(「NTSC」)編碼器或耦接至多核心CPU 110A之任何其他類型的視訊編碼器134。視訊放大器136耦接至視訊編碼器134及觸控螢幕顯示器132。視訊埠138耦接至視訊放大器136。通用串列匯流排(「USB」)控制器140耦接至CPU 110A。又，USB埠142耦接至USB控制器140。用戶識別模組(SIM)卡146亦可耦接至CPU 110A。另外，如圖1中所示，數位相機148可耦接至CPU 110A。在一例示性態樣中，數位相機148為電荷耦合裝置(「CCD」)相機或互補金氧半導體(「CMOS」)相機。

如圖1中進一步說明，立體聲音訊CODEC 150可耦接至類比信號處理器126。此外，音訊放大器152可耦接至立體聲音訊CODEC 150。在一例示性態樣中，第一立體聲揚聲

器 154 及第二立體聲揚聲器 156 耦接至音訊放大器 152。圖 1 展示麥克風放大器 158 亦可耦接至立體聲音訊 CODEC 150。另外，麥克風 160 可耦接至麥克風放大器 158。在一特定態樣中，調頻(「FM」)無線電調諧器 162 可耦接至立體聲音訊 CODEC 150。又，FM 天線 164 耦接至 FM 無線電調諧器 162。另外，立體聲頭戴式耳機 166 可耦接至立體聲音訊 CODEC 150。

圖 1 進一步指示射頻(「RF」)收發器 168 可耦接至類比信號處理器 126。RF 開關 170 可耦接至 RF 收發器 168 及 RF 天線 172。如圖 1 中所展示，小鍵盤 174 可耦接至類比信號處理器 126。又，具有麥克風之單聲道耳機 176 可耦接至類比信號處理器 126。另外，振動器裝置 178 可耦接至類比信號處理器 126。圖 1 亦展示電源供應器 180(例如，電池)耦接至晶載系統 102。在一特定態樣中，電源供應器 180 包括可再充電 DC 電池或自連接至交流(「AC」)電源之 AC 至 DC 變壓器得到的 DC 電源供應器。

如圖 1 中所描繪，觸控螢幕顯示器 132、視訊埠 138、USB 埠 142、相機 148、第一立體聲揚聲器 154、第二立體聲揚聲器 156、麥克風 160、FM 天線 164、立體聲頭戴式耳機 166、RF 開關 170、RF 天線 172、小鍵盤 174、單聲道耳機 176、振動器 178 及電源供應器 180 係在晶載系統 322 外部。

圖 2 係功能方塊圖，其說明該用於管理可攜式計算裝置中之平行資源請求之方法及系統的實例操作環境 200。該

操作環境200包括兩個處理器，在此實例中，該等處理器可為CPU 110A及資源功率管理器107。一資源(在此實施例中其可為系統匯流排207)由資源功率管理器107管理。另一資源(在此實施例中其可為用於系統匯流排207之資源代理207A)位於CPU 110A上且由該CPU 110A管理。資源代理被定義為遠端資源之本端代表，該本端代表之狀態反映遠端資源之狀態。

在另一實施例中，可由充當用戶端與資源之間的介面的構架來代替資源代理207A，該構架經實施為CPU 110A上之函式庫。下文描述此構架之實例。

如上文所描述，CPU 110A包含第0核心222、第一核心224及第N核心226。然而，CPU 110A亦可為單核心處理器，其中CPU 110A將為唯一之處理器。在以下之描述中，雖然CPU 110A包含複數個核心，但CPU 110A將被稱作單處理器(singular processor)。類似地，資源功率管理器107可包含一或多個核心。

CPU 110A上之每一核心可執行一或多個線緒。線緒係處理器上之執行單元，且在一給定核心上在任何給定之時間點僅一個線緒可為在作用中的。在一或多個線緒之內容脈絡中產生用戶端且發出請求。在下文之論述中，術語用戶端(或若干用戶端)及線緒(或若干線緒)可互換地使用。當用戶端發出請求時，其意謂在有效線緒之內容脈絡中執行的用戶端發出請求。同樣地，在用戶端在繼續進行之前等待對請求之回應的意義上，將該用戶端稱為同步的，此

意謂用戶端在其中發出請求的線緒被暫時中止直至接收到回應為止。應注意，可在單一線緒之內容脈絡中產生多個用戶端，但出於此描述之目的，在給定線緒中在給定之時間點僅一個用戶端係在作用中的。

在圖2中所示之實例中，第一用戶端201及第二用戶端202各自執行於第0核心222上。第三用戶端204經展示為執行於第一核心224上且第N用戶端206經展示為執行於第N核心226上。然而，用戶端與各個核心之耦合係任意的且係僅出於說明性目的而展示於圖2中。在CPU 110A為單核心處理器之實施例中，所有該等用戶端將執行於CPU 110A上。另外，在以下之描述中，亦可將用戶端與CPU 110A內之各個核心之間的操作互動中之任一者描述為直接發生於CPU 110A與各別用戶端之間。

將CPU 110A展示為藉由虛線211經由資源代理207A耦接至系統匯流排207，而將系統匯流排207展示為經由實線212耦接至資源功率管理器107。此連接拓撲是為了指示資源功率管理器107管理系統匯流排207，而CPU 110A具有經由資源代理207A對系統匯流排207之邏輯存取，但並不主動地管理其。

亦將資源代理207A展示為包括旗標213，該旗標213表示條件變數之狀態。條件變數表達或表示在給定之時間點可為真或可為非真的某一條件。可傳信該條件變數以指示所表達之條件在當前為真，且可等待該條件變數直至條件為真的。在此實例中，由旗標213表達之條件變數指代對匯

流排 207 之給定請求是否得到服務。當該請求得到服務時，該條件變數被傳信(CV/S)(或設定旗標 213)以指示此狀態。在那之前，該條件變數不被傳信(CV/NS)。下文將更詳細地描述旗標 213 之操作。

大體而言，將使用用戶端 201、202、204 及 206 中之任一者對一資源(在此實例中為系統匯流排 207)進行請求之實例來描述用於管理可攜式計算裝置中之平行資源請求之方法及系統。由於 CPU 110A 不主動地管理系統匯流排 207，所以其經由資源代理 207A 而將用戶端請求轉遞至管理系統匯流排 207 之處理器。在此實例中，管理系統匯流排 207 之處理器係資源功率管理器 107。因此，來自(例如)用戶端 201 之請求被從 CPU 110A 發送至資源功率管理器 107，如由線 208 指示。當該請求得到服務時，回覆(呈中斷之形式)被從資源功率管理器 107 發送至 CPU 110A，如由參考數字 209 指示。自資源功率管理器 107 發送至 CPU 110A 之中斷由 CPU 110A 上之中斷服務常式(ISR)處置。該 ISR 將傳信變數旗標 213 之條件，從而導致其將狀態自「條件變數未傳信」(CV/NS)改變至「條件變數被傳信」(CV/S)。下文將更詳細地描述中斷之影響。

將參看圖 3、圖 4、圖 5 及圖 6 來進一步描述用於管理可攜式計算裝置中之平行資源請求之方法及系統的操作。

圖 3 係說明執行於處理器(在此實例中為 CPU 110A)上之用戶端的圖。在 216 處，用戶端正在執行且發出一資源請求，該資源請求由 CPU 110A 上之資源代理 207A 轉遞至資

源功率管理器107，如由參考數字217指示。該請求可針對由資源功率管理器107管理之任何資源。在點215處，當由資源功率管理器107來處理該請求時，用戶端在CPU 110A上停止執行(或被暫時中止或封鎖)，如由參考數字218指示。一旦資源功率管理器107處理了該請求，便將應答(例如，呈中斷之形式)自資源功率管理器107發送至CPU 110A，如由參考數字219指示，藉此，在點220處，用戶端在CPU 110A上恢復執行，如由參考數字221指示。將此請求稱作同步請求，因為執行於CPU 110A上之用戶端線緒216在由資源功率管理器107處理該請求時在點215處暫時中止，且直至由CPU 110A接收到應答219時的點220才恢復。然而，存在一些個例，其中需要使用戶端在等待請求被處理之時間期間執行某一其他工作。亦有可能用戶端無需等待來自資源功率管理器107之應答且可簡單地繼續執行。可將此請求稱為「自主導引(fire-and-forget)」請求。自主導引請求之實例將為用戶端關閉一資源的請求。用戶端(且擴展而言，用戶端正在其中執行之線緒)不必等待到此動作完成才繼續。圖4中描述此方法之實例。

圖4係展示用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例的操作的圖。在圖4中，第一線緒231係在CPU 110A中開始。在點232處，來自此線緒中之用戶端的請求由資源代理207A自CPU 110A轉遞至資源功率管理器107。在一實施例中，該請求可來自(例如)請求系統匯流排207上之某一頻寬量的第一用戶端201。舉例而

言，系統匯流排207可以每秒100百萬位元(MB/s)之頻寬操作，但用戶端201將希望其以200 MB/s操作。因此，由參考數字234指示之請求係對資源功率管理器107(其管理系統匯流排207)之請求，其係請求系統匯流排207上之額外頻寬以允許第一用戶端201以更高頻寬操作。

在點232處，該請求被從CPU 110A上之資源代理207A轉遞至資源功率管理器107。然而，由於第一用戶端201不必等待直至系統匯流排207上之可用頻寬經增加以繼續操作(第一用戶端201可以較低頻寬執行直至資源功率管理器107增加系統匯流排207上可用之頻寬，在增加後，第一用戶端201自動地開始使用較高頻寬)，第一用戶端201可藉由傳入一偏好設定而向CPU 110A上之資源代理207A指示此。

該偏好設定可由第一用戶端201指定且其判定資源是否可被分叉以允許平行或非同步處理。偏好設定之實例包括(但不限於)ALLOWED(允許)、DISALLOWED(不允許)及DEFAULT(預設)。偏好設定ALLOWED(允許)及DISALLOWED(不允許)明確地允許或不允許如上文所描述之平行處理。偏好設定DEFAULT(預設)通知資源用戶端允許該資源關於是否分叉而作出其自己的決定，只要其不影響用戶端行為便可。舉例而言，若指定偏好設定DEFAULT(預設)，則資源可選擇將關閉資源或取消先前所要求之請求的所有請求分叉。關閉資源或取消先前之請求的請求通常指示用戶端對此請求後之資源狀態不感興趣且該資源可

將此請求分叉並立即傳回至用戶端。然而，若請求係針對一特定值且偏好設定為DEFAULT(預設)，則其不能將此請求分叉，因為用戶端可預期在控制傳回至用戶端之前該資源處於(至少)此特定狀態。

偏好設定由CPU 110A上之資源代理207A用來以非同步方式將請求轉遞至資源功率管理器107，使得當該請求得到服務時，第一用戶端201或該第一用戶端201在其中執行之線緒不被暫時中止。在此個例中，CPU 110A繼續可用以執行在231處開始之線緒且如由參考數字237所指示進行操作。當因此發出一資源請求但該資源請求尚未得到服務，或發出該請求且該請求得到服務但該新狀態尚未由CPU 110A上之資源代理207A應答或為用戶端可見(亦即，若用戶端將查詢該資源之狀態，則其將仍「看見」舊狀態)時，將該請求及該資源標記為「分叉」。術語「分叉」指代資源處於不相干狀態的條件(在發出了對資源之上一請求，但尚未得到服務；或若得到服務，則資源之本端代表(如由資源代理207A維護)尚未更新至新狀態的意義上)。資源代理207A亦可登記在請求實際得到服務之後將被調用的回叫。在此回叫中，資源代理207A可選擇執行取決於資源正處於新(被請求)狀態的其他動作。舉例而言，有可能在系統匯流排207之頻寬增加之後應將時脈資源(未圖示)設定至新值。在此狀況下，資源代理207A將在回叫中發出對時脈資源(未圖示)之請求，因此確保僅在對系統匯流排207之請求得到服務之後才執行對時脈資源(未圖示)

之請求。與此同時，資源功率管理器107處理該請求(如由參考數字235指示)且將應答發送至CPU 110A(指示於參考數字236處)。應答236對應於經由連接209(如圖2中所示)所傳達之中斷。CPU 110A上之處置/服務於此中斷的中斷服務常式(ISR)將與旗標213相關聯之條件變數設定至「經傳信」狀態(亦即，CV/S)。此動作指示主題資源現為「可加入的」。術語「可加入」指代資源(例如，系統匯流排207)已服務於該請求、已移至新狀態且已向請求處理器上之資源代理207A指示此新狀態的條件。當由CPU 110A上之用戶端201、202、204及206中的一者發出對系統匯流排207之後續請求時，該資源「被加入」。術語「被加入」指代資源代理207A將資源(在此實例中為系統匯流排207)之本端代表更新至新狀態，因此使該資源為相干的且能夠服務於新請求的動作。資源代理207A亦將執行其已在分叉處登記之任何回叫，如上文所描述。其將接著處置新請求。

如圖4中進一步展示，執行於線緒(指示於參考數字241處)中之第二用戶端或執行於新線緒中之同一用戶端201在點242處作出對系統匯流排207之新請求。因此，在點242處，資源(在此實例中為系統匯流排207及其本端資源代理207A)被「加入」，此係可能的，因為與旗標213相關聯之條件變數被設定(CV/S)且將資源標記為「可加入」，如上文所描述。若在點242處，當第二請求到達資源代理207A時，系統匯流排207尚未為「可加入」的，則該請求及該請求在其中執行之線緒將封鎖直至來自資源功率管理器

107之中斷將資源標記為「可加入」。此新線緒中之處理接著如由參考數字244指示而繼續。

圖5係時刻表圖，其展示用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例的操作。在時間 t_0 ，執行於CPU 110A上之第一用戶端201開始工作(如使用參考數字251所說明)。在時間 t_1 ，第一用戶端201經由本端資源代理207A而發出一針對遠端資源(諸如系統匯流排207)之請求，如指示於參考數字252處。該請求被接收於本端資源代理207A處且開始在CPU 110A上執行(如使用參考數字254所說明)。在時間 t_2 ，資源代理207A將請求轉遞至第二處理器(資源功率管理器107)，如使用參考數字255所說明。當資源功率管理器107接收到該請求時，資源功率管理器107開始執行及處理該請求，如使用參考數字256所說明。根據用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例，假定第一用戶端201允許請求之「分叉」，則在時間 t_3 ，資源代理207A以非同步方式處理該請求(亦即，藉由向遠端處理器(資源功率管理器107)發出該請求，但不等待直至該請求得到服務及應答)且傳回至第一用戶端201，如使用參考數字257所指示。第一用戶端201現繼續工作，如使用參考數字260所指示。如在時刻表中所示，由第一用戶端201在指示於260處之週期期間執行之工作係與由資源功率管理器107在由256指示之時間週期期間執行之工作平行地發生。該請求由資源代理207A傳回至第一用戶端201所處的點258被稱作「分叉」點。該分叉

點指示：在點258處，資源或請求分叉以允許同時在資源功率管理器107中及在用戶端201中之平行處理。

在時間t4，發生至第二線緒之內容脈絡切換，如使用參考數字266所指示。在時間t5，資源功率管理器107將中斷(如使用參考數字261所指示)發送至CPU 110A上之資源代理207A。CPU 110A中之處置此中斷的中斷服務常式(ISR)將條件變數旗標213(圖2)設定至「經傳信」(CV/S)狀態，如使用參考數字264所指示。在點265處，系統匯流排207被視為「可加入」且可用以處理後續請求。

在時間t7，執行於CPU 110A上之第二線緒中的第二用戶端202經由資源代理207A而發出對系統匯流排207之第二請求，如使用參考數字268所指示。自系統匯流排207之觀點，可將該請求268視為「後續」請求。當接收到該後續請求時，資源代理207A檢查並判定系統匯流排207處於可加入狀態且可被「加入」。在點270處，在第二線緒之內容脈絡中加入系統匯流排207。此意謂在點265處系統匯流排207被視為「可加入的」，但在點270處被「加入」，以便處理來自第二用戶端202之第二線緒中的後續請求268。資源(在此實例中為系統匯流排207)因此可處於三種狀態(分叉、可加入及被加入)中之一者。自資源被分叉的點起，將資源視為處於不相干狀態，此意謂請求被接收但尚未得到處理，或若得到處理，則尚未用實際(新)資源狀態來更新資源之本端代表。在此狀態下，資源不能處理後續請求。因此，在資源可處理後續請求268之前，資源被「加

入」(亦即，在點270處使其為相干的)。指示於271處之時間週期說明來自第二用戶端202之後續請求268由資源代理207A處理的時間期間。應注意，此請求亦可為「分叉的」，從而重複上文所描述之序列。

應提及的是，後續請求無需來自不同用戶端，而是在一實施例中，其亦可以後續請求之形式來自第一用戶端201。重要的是，系統匯流排207之任何用戶端可起始後續請求且因此在時間t7「加入」。在第一用戶端201之內容脈絡中(及在此實施例中，在一不同線緒中)發出的先前請求因此在後續請求之處理期間「完成」(在將本端資源代表(資源代理207A)更新至新狀態且因此使該資源(系統匯流排207)再次變得相干的意義上)，如上文所提及，該後續請求可來自CPU 110A上之任何線緒上的任何用戶端。此係在無額外管理的情況下發生，且對系統匯流排207之任何請求可在點270處使資源恢復可用性及工作。

圖6A及圖6B共同地說明流程圖，該流程圖描述用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例的操作。在區塊603中，接收針對資源之請求。在上文之實例中，由資源代理207A來接收該請求。然而，可由另一實體來接收該請求。在此實施例中，第一用戶端201進行針對由遠端處理器(諸如資源功率管理器107)管理之系統匯流排207的請求。在所描述之實例中，經由CPU 110A上之本端資源代理207A來進行該請求。

在區塊604中，判定發出請求之第一用戶端201是否希望

此請求同步地完成或第一用戶端201是否允許資源代理207A將此請求「分叉」並立即傳回至第一用戶端201。若第一用戶端201不允許分叉且想要同步地完成該資源請求，則在區塊606中，資源代理207A將請求轉遞至資源功率管理器107，封鎖處理線緒，且等待直至資源功率管理器107服務於該請求。

在區塊608中，若第一用戶端201已允許該請求分叉，則資源代理207A將該請求轉遞至資源功率管理器107且立即傳回至用戶端，而不等待該請求得到服務。

在區塊612中，控制傳回至第一用戶端201且線緒繼續執行，而不管該請求尚未由資源功率管理器107服務。該資源被標記為分叉且在其可服務於任何後續請求之前需要被加入。

在區塊614中，資源功率管理器107完成對請求之處理且將一指示此情況之中斷發送至CPU 110A。CPU 110A上之處置此中斷的中斷服務常式(ISR)傳信條件變數(旗標213設定至(CV/S))，從而導致資源被設定為「可加入」狀態。區塊614中所描述之步驟可如所說明而逐次地發生，平行地發生，或在區塊612中所描述之步驟之後且在區塊628中所描述之步驟之前的任何時間發生。

在區塊618中，起始一後續請求。可在同一線緒中或在第二線緒中起始該後續請求。另外，可由同一用戶端或由一不同用戶端來起始該後續請求。在此實例中，由一不同用戶端(亦即，不同線緒中之第二用戶端202)來起始該後續

請求。

在另一實施例中，可自與第一請求相同之線緒來起始後續請求。由於初始線緒中之處理未被第一請求封鎖，所以該初始線緒可做其他工作且接著將一後續請求置於資源上。舉例而言，一線緒可發出關閉遠端資源之請求，該遠端資源由本端資源代理分叉。在一時間週期之後，該線緒(或更具體言之，該線緒中之用戶端)可想要開啟該資源。在此狀況下，在同一線緒中起始後續請求。

在區塊620中，在資源代理207A處接收後續請求。

在區塊624中，判定資源(系統匯流排207)是否為可加入的。若在區塊624中判定資源並非可加入的，則該過程進行至區塊626，在該區塊626中，線緒等待直至資源係可加入的。若在區塊624中判定資源係可加入的或在資源在區塊626中變得可加入之後，該程序進行至區塊628，在該區塊628中，資源被「加入」，資源代理207A完成來自先前分叉之請求的任何擱置中處理且將資源之本端代表更新至新狀態。以此方式，資源被移至「加入」狀態。該資源現開始處理第二請求。

隱式加入(其在後續請求到達處於「分叉」或「可加入」狀態之資源時隨之發生)之替代例稱為「顯式加入」。如上文所描述，每當對資源之後續請求到達資源代理(或構架)時，一分叉之資源便隱式地加入。然而，可存在用戶端希望在不發出後續請求的情況下顯式地加入資源的個例。在此狀況下，將分叉之資源可藉以被加入且變為相干

的方式稱作「顯式加入」。此顯式加入操作將等待條件變數被傳信，更新本端資源代表以反映新資源狀態且傳回至用戶端。該用戶端將被暫時中止直至此呼叫傳回。

對遠端資源之「分叉請求」的一替代例係分叉之「本端」資源/請求。在上文之實例中，假定請求在遠端得到服務且用戶端(或呼叫者)在其請求得到服務時希望在請求處理器上執行其他操作或可能聽從於其他用戶端或其他線緒以執行其他操作。無需總為該情況。考慮其中請求由CPU 110A上之另一線緒來服務的情況。在此狀況下，用戶端發出對此資源之請求，該請求由另一線緒拾取(每當其由作業系統排程時)、服務且應答被傳回至呼叫線緒。有可能藉由用此請求處理線緒來代替遠端處理器而將上文之方法及系統應用於此情況。

將「加入」之一替代例稱作「積極式加入」狀態。在上文所論述之實施例中，分叉之資源在後續請求到達資源(或其代理)時被隱式加入或在用戶端請求其時被顯式加入。在另一實施例中，可獨立於後續請求或顯式加入而存在經產生以用於加入一「可加入」資源之明確目的的多個背景工作線緒。在此情況中，自遠端處理器接收到完成之通知的ISR將不僅傳信條件變數並將資源標記為「可加入」，而且將把加入動作(加入回叫)作為工作項目置於佇列上。正在佇列上等待之背景工作線緒將看見此工作項目，拾取該工作項目並在其內容脈絡中執行該工作項目；確保執行加入動作且儘可能快地將資源設定至「加入」狀態。

如上文所提及，在一替代性實施例中，用戶端與資源之間的介面可為構架而非資源代理。下文論述此構架之設計及結構。

提供如下文所描述之圖7A至圖10以描述如何建立及維持圖2之節點架構。圖7A係軟體架構500A之第一態樣的圖，該軟體架構500A用於建立及維持圖4中所說明之節點架構。

圖7A係包含表示軟體或硬體(或兩者)之功能區塊的圖。圖7A說明一架構或構架管理器440，其耦接至諸如(但不限於)以下各者之複數個硬體及軟體元件：ICB驅動器模組103；中央處理單元110，其亦大體稱作第一硬體元件(硬體元件#1)；CPU 110之時脈442，其亦大體稱作第二硬體元件(硬體元件#2)；匯流排仲裁器或排程器422，其亦大體稱作第三硬體元件(硬體元件#3)；匯流排程式A-444A，其亦大體稱作第一軟體元件(軟體元件#1)；匯流排程式B-444B，其亦大體稱作第二軟體元件(軟體元件#2)；時脈程式AHB，其大體稱作第三軟體元件(軟體元件#3)；由大體被指示為按鍵的軟體元件448監視之動作或功能；及包含軟體元件或硬體元件或兩者之舊版元件450。

舊版軟體元件之實例可包括(但不限於)動態環境管理器(DEM)。此係處置處理器睡眠事件之處理器間通知的軟體模組。舉例而言，第一處理器A使用DEM來接收第二處理器B已變成閒置/自閒置恢復的通知。在較新之硬體上，此軟體功能性已被歸入路由處理器模組(RPM)子系統/通信協

定中。存在其他舊版軟體元件且其被包括於本發明之範疇內。

舊版硬體元件之實例可包括(但不限於)AMBA(高級微控制器匯流排架構)高效能匯流排(AHB)。在較舊之PCD 100上，AHB可包含主要系統匯流排，而在較新之PCD 100上，系統匯流排組構107係完全不同的，且AHB匯流排僅用於特殊應用以與尚未被更新以經由新系統匯流排組構來通信的模組通信。存在其他舊版硬體元件且其被包括於本發明之範疇內。

構架管理器440可包含一電腦指令庫，該庫管理資料結構，諸如與上述硬體元件及軟體元件中之每一者通信的節點。構架管理器440可負責產生可形成節點602、622、642及646之一或多個資源，如在圖7A之虛線A的右側所說明。

構架管理器440可直接與駐存於CPU 110上之每一ICB驅動器模組103通信。位於圖7A之右側的每一節點602、622、642及646係位於圖7A之虛線A之左手側的每一軟體或硬體元件之代表或模型。對於本發明之剩餘部分而言，將用參考數字601來表示一般或非特定節點，如圖7B中所說明。

如先前所提及，圖7A之每一例示性節點602、622、642及646可包含一或多個資源。資源可包含軟體元件或硬體元件或兩者。舉例而言，第一節點602包含大體與第一硬體元件或中央處理單元110對應之單一資源。在本發明中

所描述之軟體架構的情況下，節點601之每一資源可具備包含一或多個文數字字元之唯一名稱。在圖7A中所說明之例示性實施例中，第一節點602之資源已指派有資源名稱「core/cpu」。此例示性資源名稱大體上對應於為一般熟習此項技術者已知之習知檔案命名結構。然而，如由一般熟習此項技術者認識到，含有文數字字元及/或符號之任何其他組合的其他類型之資源名稱完全在本發明之範疇內。

在圖7A之例示性實施例中，第二節點622包含複數個資源。具體言之，在此特定例示性實施例中，第二節點622具有第一資源，該第一資源包含對應於匯流排仲裁器或排程器422之單一硬體元件。第二節點622之第二資源包含一軟體元件，該軟體元件大體對應於匯流排程式A 444A之第一軟體元件。第二節點622之第三資源包含另一軟體元件，該軟體元件大體對應於匯流排程式B 444B之第二軟體元件。一般熟習此項技術者會認識到，一給定節點601之資源及資源類型的任何組合及任何數目的資源及資源類型係完全在本發明之範疇內。

除產生節點601外，構架管理器440亦可產生或具現化標誌650。標誌可包含一或多個舊版元件(諸如硬體元件或軟體元件(或兩者以及複數個此等元件))，該一或多個舊版元件不容易映射其自身或不易與由構架管理器440管理之軟體架構相容。標誌650可支援節點601之資源，其意謂節點601之資源可依賴於標誌650。標誌650之一實例可包括字串驅動器。字串驅動器可不容易配合於結合圖7A所描述之

架構內。標誌650可由節點601參考且其相依性陣列資料收集於圖8之區塊1125中。

圖7A亦說明第一用戶端648，其大體對應於兩個軟體元件448、450之動作或功能。在圖7A中所說明之例示性實施例中，第一用戶端648大體對應於可發生於由可攜式計算裝置100支援之特定應用程式模組105內的按鍵動作。然而，一般熟習此項技術者會認識到，除按鍵之外的軟體元件之其他動作及/或功能係完全在本發明之範疇內。下文將結合圖10來描述關於用戶端請求648及其各別產生的進一步細節。

圖7A亦說明特定架構元件之間的關係。舉例而言，圖7A說明用戶端648與第一節點602之間的關係。具體言之，第一用戶端648可產生用戶端請求675A(用虛線說明)，該用戶端請求675A由包含資源「/core/cpu」之第一節點602管理或處置。通常，存在一預定或設定數目之類型的用戶端請求675。下文將結合圖10來進一步詳細描述用戶端請求675。

圖7A中所顯示之其他關係包括用虛線680說明之相依性。相依性係另一節點601之各別資源之間的關係。相依性關係通常指示第一資源(A)依賴於可向該第一資源(A)提供資訊的第二資源(B)。此資訊可為由第二資源(B)執行之操作的結果或其可僅包含第一資源(A)所需要之狀態資訊或其任何組合。第一資源(A)及第二資源(B)可為同一節點601之部分或其可為不同節點601之部分。

在圖7A中，第一節點602依賴於第二節點622，如由相依性箭頭680B指示，該相依性箭頭680B由第一節點602發起且延伸至第二節點622。圖7A亦說明第一節點602亦依賴於第三節點642，如由相依性箭頭680A所說明。圖7A亦說明第二節點622依賴於第四節點646，如由相依性箭頭680C所說明。一般熟習此項技術者會認識到，用圖7A之虛線箭頭說明之相依性680實質上僅為例示性的且各別節點601之間的相依性之其他組合係在本發明之範疇內。

架構或構架管理器440負責維持上文所描述之關係，該等關係包括(但不限於)圖7A中所說明之用戶端請求675及相依性680。構架管理器440將試圖具現化或產生與其可具現化或產生的節點的一樣多的節點601，只要任一給定節點601之相依性680係完全的便可。當支援相依性之資源存在或處於準備好處置與相依性680有關之資訊的狀態時，相依性680係完全的。

舉例而言，若包含單一資源「/clk/cpu」之第三節點642尚未產生，則包含單一資源「/core/cpu」之第一節點602可未由構架管理器440產生或建立(由於存在於第一節點602與第三節點642之間的相依性關係680A)。一旦第三節點642已由構架管理器440產生，則構架管理器440便可由於相依性關係680A而產生第二節點602。

若構架管理器440未能產生或具現化一特定節點601(因為其相依性680中之一或多者係不完全的)，則構架管理器440將繼續執行或執行對應於由構架管理器440成功地產生

之彼等節點 601 的步驟。構架管理器 440 將通常跳過針對一特定節點 601 (該特定節點 601 可歸因於不完全相依性 (其中依賴之資源尚未產生) 而不存在) 的呼叫，且將反映彼不完全狀態之訊息傳回至彼呼叫。

在多核心環境 (諸如圖 1 中所說明) 中，構架管理器 440 可在單獨之核心 (如圖 1 之第 0 核心 222、第一核心 224 及第 N 核心 226) 上產生或具現化節點 601。可大體在多核心環境中在單獨之核心上平行地產生節點 601，只要該等節點 601 並非依賴於彼此及一特定節點之所有相應相依性 (如下文所描述) 係完全的便可。

圖 7B 係系統之軟體架構 500B1 之第二態樣的總圖，該系統管理圖 1 之 PCD 100 的資源。在此總圖中，每一節點 601 之一或多個資源尚未具備唯一名稱。圖 7B 之節點或資源圖表 500B1 僅包含節點 601、標誌 650、用戶端 648、事件 690 及由架構或構架管理器 440 支援之查詢函式 695。每一節點 601 經說明為具有橢圓形狀及具有特定方向 (其表示節點 601 內之資源之間的各別相依性) 之箭頭 680。

可使在圖 7A 至圖 7B 內說明之節點架構內的呼叫針對節點 601 內之資源的別名或實際資源名稱。根據一例示性實施例，不存在用以依靠標誌 650 來作出用戶端請求 675 的方法，因為在用戶端 648 與標誌 650 之間不存在介面，因此此大體意謂與標誌 650 交換之資訊通常發起自節點 601 或資源而非用戶端 648。

舉例而言，圖 7B 之第一節點 601A 具有相依性箭頭 680A

以指示第一節點601A依賴於第二節點601B之兩個資源(資源#2及#3)。類似地，第一節點601A具有相依性箭頭680B以指示第一節點601A亦依賴於第一標誌650，該第一標誌650通常包含硬體或軟體或其組合之舊版元件。

圖7B亦說明第一節點601A之用戶端648可如何向第一節點601A發出用戶端請求675。在發出此等用戶端請求675之後，第二節點601B可觸發事件690或提供對查詢695之回應，其中對應於事件690及查詢695之訊息流回至用戶端648。

圖7C係一系統之軟體架構500B2之第二態樣的特定圖，該系統管理圖1之PCD 100的資源。圖7C說明節點或資源圖表500B2，其僅包含具有特定、但為例示性之資源名稱的節點601以及對應於圖7A之彼等用戶端、事件及查詢函式的用戶端648、事件690及查詢函式695。每一節點601經說明為具有橢圓形狀及具有特定方向(其表示節點601內之資源之間的各別相依性)之箭頭680。

舉例而言，第一節點602具有相依性箭頭680B以指示第一節點602依賴於第二節點622之三個資源。類似地，包含第二軟體元件444B且在圖7C中大體用參考字母「C」表示的第三資源「/bus/ahb/sysB/」具有相依性箭頭680C，該相依性箭頭680C指示此第三資源(C)依賴於第四節點646之單一「/clk/sys/ahb」資源。

圖7C亦說明來自節點601之輸出資料，其可包含一或多個事件690或查詢函式695。查詢函式695類似於事件690。

查詢函式695可具有一查詢控制代碼，其可為唯一的或可並非唯一的。查詢函式大體並非在外部識別且大體而言其不具有狀態。查詢函式695可用以判定節點601之特定資源的狀態。查詢函式695及事件690可與所建立之用戶端648具有關係且此等關係由定向箭頭697表示以指示來自各別事件690及查詢函式695之資訊被傳遞至特定用戶端648。圖7C亦經由相依性箭頭680D來說明圖7C之第二節點622如何依賴於第一標誌650。

圖7B至圖7C之節點或資源圖表500B表示存在於記憶體中且由構架管理器440管理的關係及可包含節點601之相關資料結構。可由構架管理器440自動地產生節點或資源圖表500B以作為用於識別由構架管理器440管理之各別元件之間的關係及用於由軟體小組進行故障診斷的有用工具。

圖7D係流程圖，其說明一種用於產生一用於管理PCD 100之資源之軟體架構的方法1000A。區塊1005係用於管理PCD 100之資源之方法或程序1000的第一常式。在常式區塊1005中，可由構架管理器440執行或運作一常式以用於接收節點結構資料。該節點結構資料可包含一概述特定節點601可與其他節點601所具有之相依性的相依性陣列。下文將結合圖8來更詳細地描述關於節點結構資料及此常式或子方法705的進一步細節。

緊接著，在區塊1010中，構架管理器440可檢閱為區塊1005中所接收之節點結構資料之一部分的相依性資料。在決策區塊1015中，構架管理器440可判定節點結構資料是

否定義葉節點601。葉節點601大體上意謂待基於節點結構資料產生之節點不具有任何相依性。若對決策區塊1015之詢問係肯定的(意謂用於產生當前節點之節點結構資料不具有任何相依性)，則構架管理器440繼續至常式區塊1025。

若對決策區塊1015之詢問係否定的，則遵循「否」分支而至決策區塊1020，在該決策區塊1020中，構架管理器判定節點結構資料內之所有硬相依性是否存在。硬相依性可包含其中資源在無其的情況下不能存在的相依性。同時，軟相依性可包含其中資源可將相依之資源用作可選步驟的相依性。軟相依性意謂即使當軟相依性不存在時，仍可在節點架構內產生或具現化具有軟相依性之節點601或節點601之資源。可將標誌650參考作為如上文所描述之軟相依性。

軟相依性之實例可包含一最佳化特徵，該最佳化特徵對於含有多個資源之資源導向節點601的操作而言並非至關重要的。構架管理器440可針對所存在之所有硬相依性來產生或具現化一節點或一資源(即使當對於具有未產生之軟相依性的彼等節點或資源而言不存在軟相依性時)。一回叫特徵可用以參考軟相依性，使得當軟相依性變得可用於構架管理器440時，構架管理器440將通知參考軟相依性之每一回叫：該等軟相依性現在係可用的。

若對決策區塊1020之詢問係否定的，則遵循「否」分支而至區塊1027，在該區塊1027中，節點結構資料由構架管

理器 440 儲存於臨時儲存器(諸如記憶體)中且構架管理器 440 產生與此未經具現化之節點相關聯的回叫特徵。

若對決策區塊 1015 之詢問係肯定的，則遵循「是」分支而至常式 1025，在該常式 1025 中，基於常式區塊 1005 中所接收之節點結構資料來產生或具現化節點 601。下文將結合圖 9 來描述常式區塊 1025 之進一步細節。緊接著，在區塊 1030 中，構架管理器 440 使用其唯一之資源名稱來公佈新近產生之節點 601，使得其他節點 601 可將資訊發送至新近產生之節點 601 或自新近產生之節點 601 接收資訊。

現參看圖 7E，圖 7E 係圖 7D 之延續流程圖，在區塊 1035 中，構架管理器 440 通知依賴於新近產生之節點 601 的其他節點 601：該新近產生之節點 601 已被具現化且準備好接收或傳輸資訊。根據一例示性態樣，當產生一相依節點(如圖 7B 之節點 601B)時立即觸發通知，亦即，遞歸地執行該等通知。因此，若建構了圖 7B 之節點 601B，則立即通知節點 601A。此通知可允許建構節點 601A(因為節點 601B 係節點 601A 之最終相依方)。節點 601B 之建構可導致其他節點 601 被通知，等等。直至依賴於節點 601B 之最終資源係完全的，節點 601B 才變成完全的。

一第二、稍微更複雜之實施係將所有通知置於一單獨之通知佇列上，且接著在單一之時間點開始來穿過該佇列(亦即，迭代地執行該等通知)。因此當建構圖 7B 之節點 601B 時，將至節點 601A 之通知推送到清單上。接著執行彼清單且通知節點 601A。此導致至其他額外節點 601(除節

點 601A 之外的節點，未說明於圖 7B 中)的通知被置於同一清單上，且接著在發送至節點 601A 之通知之後發送彼通知。至其他節點 601 之通知(除至節點 601A 之通知之外)直至在已完成與節點 601B 及節點 601A 相關聯之所有工作之後才發生。

邏輯上，此等兩個實施係等效的，但其在實施時具有不同之記憶體消耗性質。遞歸式實現係簡單的，但可消耗任意量之堆疊空間，其中該堆疊消耗係隨相依性圖表之深度而定。迭代式實施係稍微更複雜的且需要稍多的靜態記憶體(通知清單)，但堆疊使用係恆定的而無關於相依性圖表之深度，諸如圖 7B 中所說明。

又，區塊 1035 中之節點產生的通知並不限於其他節點。其亦可在內部用於別名建構。系統 500A 中之任何任意元件，而非僅其他節點，可使用相同之機制以在一節點(或標誌)變得可用時請求通知。節點與非節點兩者可使用相同之通知機制。

在決策區塊 1040 中，構架管理器 440 判定現在是否基於當前節點 601 之產生而釋放了其他節點 601 或軟相依性以用於產生或具現化。決策區塊 1040 大體上判定是否可因為某些相依性關係 680 已由近來經歷產生或具現化之當前節點實現而產生資源。

若對決策區塊 1040 之詢問係肯定的，則遵循「是」分支而返回至常式區塊 1025，在該常式區塊 1025 中，由於由剛產生之節點 601 實現了相依性，所以現可產生或具現化被

釋放之節點601。

若對決策區塊1040之詢問係否定的，則遵循「否」分支而至區塊1045，在該區塊1045中，構架管理器440可管理如圖2中所說明之架構之元件之間的通信。緊接著，在區塊1050中，構架管理器440可繼續藉由使用與特定資源相關聯之資源名稱而將由資源採取之動作記入日誌或進行記錄。在由構架管理器440或由構架管理器440管理之元件中之任一者(諸如資源、節點601、用戶端648、事件695及查詢函式697)採取的任何動作之後，可由構架管理器440執行區塊1045。區塊1045展示節點架構之另一態樣，其中構架管理器440可維護一執行中之活動日誌，其根據由產生特定元件(諸如節點601之資源)之作者提供的元件的唯一識別符或名稱來列出由每一元件執行之動作。

與先前技術相比較，區塊1050中之對活動的此日誌記錄(其列出指派給系統之每一資源的唯一名稱)係唯一的且可提供顯著優點(諸如用於除錯及錯誤故障診斷)。節點架構500A之另一唯一態樣係單獨之小組可致力於彼此獨立之不同硬體及/或軟體元件，其中每一小組將能夠使用資源名稱(其係唯一的且容易追蹤)而無需產生表來轉譯由其他小組及/或原始設備製造商(OEM)指派之較無意義且通常混亂的資源名稱。

緊接著，在決策區塊1055中，構架管理器440判定是否已請求由構架管理器440記錄之活動的日誌。若對決策區塊1055之詢問係否定的，則遵循「否」分支而至該程序之

結尾，其中該程序返回至常式1005。若對決策區塊1055之詢問係肯定的，則遵循「是」分支而至區塊1060，在該區塊1060中，構架管理器440將包含有意義之資源名稱及由該等資源名稱執行之各別動作的活動日誌發送至一輸出裝置(諸如印表機或顯示螢幕及/或兩者)。該程序接著返回至上文所描述之常式區塊1005。

圖8係流程圖，其說明圖7D之用於在PCD 100中之軟體架構中接收節點結構資料的子方法或常式1005。區塊1105係圖7D之子方法或常式1005中的第一步驟。在區塊1105中，構架管理器440可接收軟體或硬體元件(諸如CPU 110及圖7D之時脈442)之唯一名稱。如先前所論述，節點601必須參考至少一資源。每一資源在系統500A中具有唯一名稱。可用唯一名稱來識別系統500A內之每一元件。自字元觀點，每一元件具有唯一名稱。換言之，大體而言，在系統500A內不存在具有相同名稱之兩個元件。根據系統之例示性態樣，節點601之資源可大體上跨越系統具有唯一名稱，但不要求用戶端或事件名稱係唯一的(雖然其可按需要為唯一的)。

出於方便，可使用使用正斜線「/」字元的習知樹檔案命名結構或檔案命名「象徵」來產生唯一名稱，諸如(但不限於)CPU 110之「/core/cpu」及時脈442之「/clk/cpu」。然而，如由一般熟習此項技術者會認識到，含有文數字字元及/或符號之任何其他組合的其他類型之資源名稱係完全在本發明之範疇內。

緊接著，在區塊 1110 中，構架管理器 440 可接收與所產生之節點 601 之一或多個資源相關聯之一或多個驅動器函式的資料。驅動器函式大體包含待由用於特定節點 601 之一或多個資源完成的動作。舉例而言，在圖 7A 至圖 7B 中，用於節點 602 之資源 /core/cpu 的驅動器函式可請求其為了提供已被請求之所請求處理量所需之匯流排頻寬量及 CPU 時脈頻率。此等請求將經由節點 642 及節點 622 中之資源的用戶端(諸如圖 2 中之用戶端 201、202、204 及 206)而作出。用於節點 642 中之 /clk/cpu 的驅動器函式將通常負責根據其自節點 602 之 /core/cpu 資源接收之請求來實際上設定實體時脈頻率。

在區塊 1115 中，構架管理器 440 可接收節點屬性資料。該節點屬性資料大體上包含定義諸如以下各者之節點策略的資料：安全性(可經由使用者空間應用程式來存取節點？)、遠端性(可自系統中之其他處理器來存取節點？)及可存取性(資源可支援多個同時之用戶端？)。構架管理器 440 亦可定義允許資源超越預設之構架行為的屬性，諸如請求評估或日誌記錄策略。

隨後，在區塊 1120 中，構架管理器 440 可接收針對所產生之特定節點 601 而定製的使用者資料。該使用者資料可包含空「星形」欄位，如由一般熟習此項技術者關於「C」程式設計語言所理解。使用者資料亦被一般熟習此項技術者已知為「信任我」欄位。例示性定製之使用者資料可包括(但不限於)諸如頻率表之表、暫存器映射等。在

區塊 1120 中所接收之使用者資料未由系統 500A 參考，而是允許資源之定製(若該定製未由構架管理器 440 辨別或完全支援)。此使用者資料結構係「C」程式設計語言中之意欲經擴展以用於特殊或特定用途的基底類別。

一般熟習此項技術者會認識到，用於擴展一特定類別之特定用途的其他種類之資料結構係在本發明之範疇內。舉例而言，在「C++」(C-加-加)程式設計語言中，一等效結構可包含關鍵字「公共」，其將變成用於節點 601 內之資源的擴展機制。

緊接著，在區塊 1125 中，構架管理器 440 可接收相依性陣列資料。該相依性陣列資料可包含經產生之節點 601 所依賴於之一或多個資源 601 的唯一及特定名稱。舉例而言，若產生圖 7C 之第一節點 602，則在此區塊 1125 中，相依性陣列資料可包含第二節點 622 之三個資源的資源名稱及第一節點 602 所依賴於之第三節點 642 的單一資源名稱。

隨後，在區塊 1130 中，構架管理器 440 可接收資源陣列資料。該資源陣列資料可包含所產生之當前節點的參數(諸如在產生圖 7B 至圖 7C 之第一節點 602 的情況下與此第一節點 602 相關的參數)。該資源陣列資料可包含以下資料中之一或多者：其他資源之名稱；單位；最大值；資源屬性；插入式資料；及類似於區塊 1120 之定製使用者資料的任何定製之資源資料。插入式資料大體上識別自軟體程式庫擷取之函式且通常列出可由所產生之特定節點或複數個節點支援的用戶端類型。插入式資料亦允許對用戶端產生

及消滅的定製。在區塊1130之後，該程序返回至圖7D之區塊1010。

在圖8中，已用虛線說明屬性資料區塊1115、定製使用者資料區塊1120及相依性陣列資料區塊1125以指示此等特定步驟係可選的且對於任一給定節點601而言係不需要的。同時，已用實線說明唯一名稱區塊1105、驅動器函式區塊1110及資源陣列資料區塊1130以指示常式1005之此等步驟大體上對產生節點601而言係強制性的。

圖9係流程圖，其說明圖7D之用於在PCD 100之軟體架構中產生節點的子方法或常式1025。根據一例示性實施例，常式區塊1205係用於具現化或產生節點601之子方法或常式1025中的第一常式。在常式區塊1205中，與被具現化之節點601相關聯的一或多個用戶端648係產生於此步驟中。下文將結合圖10來進一步詳細描述關於常式區塊1205之進一步細節。

在區塊1210中，構架管理器可產生或具現化對應於區塊705之節點結構資料的一或多個資源。緊接著，在區塊1215中，構架管理器440可啟動在常式區塊1005之常式區塊1110中接收的驅動器函式。根據一例示性態樣，可使用在常式區塊1005之資源陣列資料區塊1130中接收的最大值來啟動該等驅動器函式。根據另一較佳之例示性態樣，可用與來自常式1005之節點結構資料一起傳遞的可選初始值來啟動每一驅動器函式。若未提供初始資料，則將驅動器函式初始化於0(最小值)。亦通常以一方式來啟動驅動器函

式使得已知其正被初始化。此使得資源能夠執行係特定針對初始化但無需在正常或常式操作期間執行的任何操作。該程序接著返回至圖7D之步驟1030。

圖10係流程圖，其說明圖9之用於在PCD 100之軟體架構中產生用戶端648的子方法或常式1205。區塊1305係常式區塊1205之第一步驟，在該區塊1305中，產生一或多個資源601之用戶端648。在區塊1205中，構架管理器440接收被指派給所產生之用戶端648的名稱。類似於資源名稱，用戶端648之名稱可包含任何類型之文數字及/或符號。

緊接著，在區塊1310中，若存在針對所產生之此用戶端648的任何特定定製，則可由構架管理器440接收定製之使用者資料。已用虛線來說明區塊1310以指示該步驟係可選的。區塊1310之定製之使用者資料類似於上文結合節點601之資源產生所論述的定製之使用者資料。

在區塊1315中，構架管理器440接收被指派給所產生之特定用戶端的用戶端類型分類。到此文章時為止，用戶端類型分類可包含四種類型中之一者：(a)必需；(b)脈衝；(c)向量及(d)等時。可取決於由系統101管理之資源及依賴於節點601之資源的應用程式來擴展用戶端類型分類清單。

必需分類大體與自必需之用戶端648傳遞至特定資源601之純量值的處理對應。舉例而言，一必需之請求可包含每秒幾百萬個指令(MIP)。同時，脈衝分類大體上對應於用以在未指定任何開始時間或停止時間的某一時間週期內完

成某一活動的請求的處理。

等時分類大體上與對通常重新發生且具有明確之開始時間及明確之結束時間的動作的請求對應。向量分類大體上與一資料陣列對應，該資料陣列通常係逐次或平行的所需多個動作的一部分。

隨後，在區塊1320中，構架管理器440接收指示用戶端648是否已被指定為同步或非同步的資料。同步用戶端648係通常要求構架管理器440鎖定節點601之資源直至資源601傳回資料及資源601已完成來自同步用戶端648之所請求任務的指示為止的用戶端。

另一方面，非同步用戶端648可由平行之一或多個線緒來處置，該一或多個線緒由構架管理器440存取。構架管理器440可產生至線緒之回叫且當已由各別線緒執行該回叫時可傳回一值。

在區塊1320之後，在決策區塊1325中，構架管理器440判定由用戶端645識別之資源是否可用。若對決策區塊1325之詢問係否定的，則遵循「否」分支而至區塊1330，在該區塊1330中，將一指示此時不能產生用戶端648的空值或訊息傳回至使用者。

若對決策區塊1325之詢問係肯定的，則遵循「是」分支而至決策區塊1335，在該決策區塊1335中，構架管理器440判定由用戶端648識別之每一資源是否支援區塊1310所提供之用戶端類型。若對決策區塊1335之詢問係否定的，則遵循「否」分支而返回至區塊1330，在該區塊1330

中，傳回一指示此時不能產生用戶端648的空值或訊息。

若對決策區塊1335之詢問係肯定的，則遵循「是」分支而至區塊1340，在該區塊1340中，構架管理器440在記憶體中產生或具現化用戶端648。緊接著，在區塊1345中，若在區塊1310中接收到任何定製之使用者資料(諸如可選引數)，則可用此等可選引數之各別資源及特定節點601來映射該等可選引數。緊接著，在區塊1350中，將新近產生之用戶端645耦接至在閒置狀態下或在所請求之狀態(如在上文所描述之圖10C中所說明)下的其相應之一或多個資源。該程序接著返回至圖12之區塊1210。

圖11係流程圖，其說明一種用於在PCD 100之軟體架構中產生針對資源601的用戶端請求675的方法1400。該方法1400係大體上在如上文結合圖7D至圖7E及圖10所描述之用戶端產生及節點產生之後執行。

區塊1405係用於產生針對資源601的用戶端請求675之方法1400中的第一步驟。此方法1400將描述如何由構架管理器440來處置以下三種類型之請求675：(a)必需；(b)脈衝；及(c)向量。如上文所提及之請求675的名稱所暗示，用戶端請求675大體上與所產生及上文描述之用戶端類型對應。

在區塊1405中，構架管理器440可接收與一特定用戶端請求675相關聯之資料，諸如上文所提及之三種類型中的一者：(a)必需；(b)脈衝；及(c)向量。與必需之請求相關聯的資料大體上包含自必需之用戶端648傳遞至一特定資

源601的純量值。舉例而言，必需之請求可包含每秒幾百萬個指令(MIP)。同時，脈衝請求包含用以在未指定任何開始時間或停止時間的某一時間週期內完成某一活動的請求。向量請求之資料大體上包含需要逐次地或平行地完成之多個動作之陣列。向量請求可包含任意長度之值。向量請求通常具有一大小值及值之陣列。節點601之每一資源可經擴展而具有指標欄位以便支援向量請求。在「C」程式設計語言中，如由一般熟習此項技術者所理解，指標欄位由聯集函式(union function)支援。

緊接著，在區塊1410中，構架管理器440經由用戶端648發出請求，該用戶端648係藉由上文結合圖10所描述之方法而產生。隨後，在區塊1415中，若該請求係必需類型或向量類型，則構架管理器440雙重緩衝經由用戶端所傳遞之請求資料。若該請求係脈衝類型，則構架管理器440跳過區塊1415。

對於必需請求而言，在此區塊1415中，來自先前請求之值被維持於記憶體中，使得構架管理器440可判定在當前組之所請求值中是否存在先前請求之值之間的任何差異。對於向量請求而言，先前請求通常未維持於記憶體中(雖然節點601之資源可按特定實施之需要維持其)。因此，區塊1415對於向量類型之請求而言係可選的。

在區塊1420中，構架管理器440在當前組之所請求值中計算先前組之所請求值之間的 Δ 或差異。在決策區塊1425中，構架管理器判定當前組之所請求值是否與先前組之所

請求值相同。換言之，構架管理器440判定在當前組之所請求值與先前組之所請求值之間是否存在差異。若在當前組之所請求值與先前組之所請求值之間不存在差異，則遵循「是」分支(其跳過區塊1430至區塊1470)而至區塊1475，在該區塊1475中，該程序結束。

若對決策區塊1425之詢問係否定的(意謂該組所請求值相對於該組預先所請求值為不同的)，則遵循「否」分支而至決策區塊1430。

在決策區塊1430中，構架管理器440判定當前請求是否為非同步請求。若對決策區塊1430之詢問係否定的，則遵循「否」分支而至區塊1440，在該區塊1440中，構架管理器440鎖定對應於用戶端請求675之資源601。若對決策區塊1430之詢問係肯定的(意謂當前請求係非同步請求類型)，則遵循「是」分支而至區塊1435，在該區塊1435中，可將請求推送至另一線緒上且可由另一核心來執行該請求(若當前構架管理器440管理多核心系統(如圖1之多核心系統))。已用虛線說明區塊1435以指示在PCD 100為單核心中央處理系統的情況下此步驟可為可選的。

隨後，在區塊1440中，構架管理器440鎖定對應於請求675之資源601。緊接著，在區塊1445中，資源601執行更新函式，該更新函式大體對應於在圖8之區塊1130中接收之資源陣列資料的插入式資料。該更新函式大體上包含負責根據新用戶端請求實現新資源狀態的函式。該更新函式將其先前狀態與用戶端請求中之所請求狀態相比較。若所

請求狀態大於先前狀態，則更新函式將執行用戶端請求。然而，若所請求狀態等於或小於當前狀態及資源操作所處之狀態，則將不執行用戶端請求以便增加效率，因為舊狀態達成或滿足所請求狀態。更新函式自用戶端獲得新請求且將其與所有其他作用中請求彙總以判定資源之新狀態。

作為一實例，多個用戶端可請求匯流排時脈頻率。用於匯流排時脈之更新函式將通常取所有用戶端請求之最大值，且將其用作匯流排時脈之新的所要狀態。雖然存在一些將由多個資源使用之更新函式，但並非為所有資源將使用相同之更新函式的狀況。一些普通之更新函式將取用戶端請求之最大值，將取用戶端請求之最小值及將用戶端請求求和。或者，若資源需要以某種獨特之方式彙總請求，則資源可定義其自己之定製更新函式。

緊接著，在區塊1450中，構架管理器440將資料傳遞至對應於用戶端648之資源，使得該資源可執行特定針對節點601之資源的驅動器函式。驅動器函式應用如由更新函式計算的資源狀態。此可需要更新硬體設定、向相依之資源發出請求、呼叫舊版函式或上述步驟之某一組合。

在先前實例中，更新函式計算所請求之匯流排時脈頻率。驅動器函式可接收彼請求之頻率且其可更新時脈頻率控制HW以在彼頻率下執行。應注意，有時不可能使驅動器函式滿足更新函式所計算之確切的所請求狀態。在此狀況下，驅動器函式可選擇最佳地滿足該請求之頻率。舉例而言，匯流排時脈HW可僅能夠在128 MHz及160 MHz下執

行，但所請求狀態可能為150 MHz。在此狀況下，驅動器函式應在160 MHz下執行，因為160 MHz超過所請求狀態。

緊接著，在區塊1455中，構架440自己在區塊1450中執行驅動器函式的資源接收狀態控制。隨後，在區塊1460中，若相對於資源而定義了事件690，則可觸發事件690使得資料被傳遞返回至對應於事件690之用戶端648。可在另一線緒中處理事件。此可最小化在資源被鎖定的情況下所花費之時間量且允許多核心系統(如圖1中所說明)中之平行操作。可以類似於可如何相對於資源來定義請求(如此方法1400中所描述)的方式而相對於資源來定義一或多個事件690。換言之，事件產生程序可很大程度上與用戶端產生程序平行。事件的一個不同之處在於有可能定義僅在某些臨限值被跨過時觸發的事件。

此種僅基於臨限值被觸發之事件的定義允許通知資源何時被過多訂購(其具有比其可支援之使用者多的同時使用者)(此指示系統過載條件)或資源何時變少/消失(此可允許其他事物被關閉、恢復在系統變得過多訂購時被禁用的功能性，等等)。由於可藉由臨限值來完成事件登記，所以其將系統在事件通知時不得不做的工作量減少為僅在確實需要做某事時才發生。亦有可能在每一狀態改變時登記事件。

緊接著，在可選區塊1465中，若所處理之請求係向量請求，則通常執行此可選區塊1465。可選區塊1465大體上包

含一用以評估向量指標是否仍定位於使用者傳遞至向量中之相同資料上的檢查或判定。若對此可選區塊1465之詢問係肯定的(意謂指標仍指向由使用者傳遞至向量中之相同資料)，則清除該指標使得不維持對舊資料之參考。大體上執行此可選區塊1465以解決在處理向量請求時上文所描述之雙重緩衝區塊1415(與脈衝請求及必需請求相比較)。

隨後，在區塊1470中，構架440解除對所請求之資源的鎖定，使得可由特定節點601之當前但現被釋放之所請求資源來處置其他用戶端請求648。該程序接著返回至第一區塊1405以用於接收下一用戶端請求。

鑒於以上揭示內容，一般熟習程式設計者能夠基於(例如)此說明書中之流程圖及相關聯之描述毫無困難地撰寫電腦程式碼或識別適當硬體及/或電路以實施所揭示之本發明。因此，並不將特定程式碼指令集或詳細硬體裝置之揭示內容視為對於充分理解如何製造及使用本發明為必要的。在以上描述中且結合可說明各種程序流程之諸圖更詳細地解釋所主張之電腦實施程序之發明性功能。

在一或多個例示性態樣中，所描述之功能可實施於硬體、軟體、韌體或其任何組合中。若以軟體實施，則該等功能可作為一或多個指令或程式碼在一電腦可讀媒體上儲存或傳輸。電腦可讀媒體包括電腦儲存媒體與通信媒體(包括促進電腦程式自一處轉移至另一處的任何媒體)兩者。儲存媒體可為可由電腦存取之任何可用媒體。藉由實例而非限制，此等電腦可讀媒體可包含RAM、ROM、

EEPROM、CD-ROM或其他光碟儲存器、磁碟儲存器或其他磁性儲存裝置，或可用以載運或儲存呈指令或資料結構之形式的所要程式碼且可由電腦存取的任何其他媒體。

又，將任何連接恰當地稱為電腦可讀媒體。舉例而言，若使用同軸電纜、光纖纜線、雙絞線、數位用戶線(「DSL」)或無線技術(諸如，紅外線、無線電及微波)而自網站、伺服器或其他遠端源傳輸軟體，則同軸電纜、光纖纜線、雙絞線、DSL或無線技術(諸如，紅外線、無線電及微波)包括於媒體之定義中。

如本文中所使用，磁碟及光碟包括緊密光碟(「CD」)、雷射光碟、光碟、數位影音光碟(「DVD」)、軟性磁碟及藍光(blue-ray)光碟，其中磁碟通常以磁性之方式再生資料，而光碟藉由雷射以光學之方式再生資料。以上各物之組合亦應包括於電腦可讀媒體之範疇內。

雖然已詳細說明及描述所選之態樣，但應理解，可在不背離如由以下申請專利範圍定義之本發明之精神及範疇的情況下在該等態樣中作出各種替代及變更。

【圖式簡單說明】

圖1係功能方塊圖，其說明一用於管理可攜式計算裝置(「PCD」)中之平行資源請求之系統的例示性元件；

圖2係功能方塊圖，其說明該用於管理可攜式計算裝置中之平行資源請求之方法及系統的一實例操作環境；

圖3係說明執行於單一資源上之同步線緒的圖；

圖4係展示用於管理可攜式計算裝置中之平行資源請求

之方法及系統的實施例的操作的圖；

圖5係時刻表圖，其展示用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例的操作；

圖6A及圖6B共同地說明流程圖，該流程圖描述用於管理可攜式計算裝置中之平行資源請求之方法及系統的實施例的操作；

圖7A係一節點架構之第一態樣的圖，該節點架構管理圖1之可攜式計算裝置之資源；

圖7B係節點架構之第二態樣的總圖，該節點架構管理圖1之可攜式計算裝置之資源；

圖7C係節點架構之第二態樣的詳圖，該節點架構管理圖1之可攜式計算裝置之資源；

圖7D係流程圖，其說明一種用於產生一用於管理可攜式計算裝置之資源之節點架構的方法；

圖7E係圖7D之延續流程圖，其說明一種用於產生一用於管理可攜式計算裝置之資源之節點架構的方法；

圖8係流程圖，其說明圖7D之用於在可攜式計算裝置中之軟體架構中接收節點結構資料的子方法或常式；

圖9係流程圖，其說明圖7D至圖7E之用於在可攜式計算裝置之軟體架構中產生節點的子方法或常式；

圖10係流程圖，其說明圖9之用於在可攜式計算裝置之軟體架構中產生用戶端的子方法或常式；及

圖11係流程圖，其說明一種用於在可攜式計算裝置之軟體架構中產生針對資源的用戶端請求的方法。

【主要元件符號說明】

100	可攜式計算裝置(PCD)
102	常式區塊
103	ICB驅動器模組
103A1	UL ICB驅動器
103A2	UL ICB驅動器
103B	LL ICB驅動器
105A1	應用程式模組1
105A2	應用程式模組2
107	資源功率管理器
110	中央處理單元(CPU)
110A	CPU
110B	圖形處理器
112	記憶體
126	類比信號處理器
128	顯示控制器
130	觸控螢幕控制器
132	顯示器/觸控螢幕
134	視訊編碼解碼器
136	視訊放大器
138	視訊埠
140	USB控制器
142	USB埠
146	SIM卡

148	CCD/CMOS相機
150	立體聲音訊編碼解碼器
152	音訊放大器
154	立體聲揚聲器
156	立體聲揚聲器
158	麥克風放大器
160	麥克風
162	FM無線電調諧器
164	FM天線
166	立體聲頭戴式耳機
168	RF收發器
170	RF開關
172	RF天線
174	小鍵盤
176	具有麥克風之單聲道耳機
178	振動器
180	電源供應器
200	操作環境
201	第一用戶端
202	第二用戶端
204	第三用戶端
206	第N用戶端
207	系統匯流排
207A	資源代理

208	線
209	參考數字
211	虛線
212	實線
213	旗標
215	點
216	用戶端線緒
217	參考數字
218	參考數字
219	參考數字
220	點
220	點
221	參考數字
222	第零核心
224	第一核心
226	第N核心
231	第一線緒
232	點
234	參考數字
235	參考數字
236	參考數字
237	參考數字
241	參考數字
242	點

244	參考數字
251	參考數字
252	參考數字
254	參考數字
255	參考數字
256	參考數字
257	參考數字
258	點
260	參考數字
261	參考數字
264	參考數字
265	點
266	參考數字
268	參考數字
270	點
271	時間週期
422	匯流排仲裁器
440	架構/構架管理器
442	CPU之時脈
444A	匯流排程式A
444B	匯流排程式B
446	時脈程式AHB
448	軟體元件
450	舊版元件

500A	系 統
500B1	軟 體 架 構
500B2	軟 體 架 構
601A	節 點
601B	節 點
602	節 點
622	第 二 節 點
642	節 點
646	節 點
648	第 一 用 戶 端
650	標 誌
675	用 戶 端 請 求
675A	用 戶 端 請 求
680A	相 依 性 箭 頭
680B	相 依 性 箭 頭
680C	相 依 性 箭 頭
680D	相 依 性 箭 頭
690	事 件
690A	事 件
690B	事 件
695	查 詢 函 式
697	定 向 箭 頭

七、申請專利範圍：

1. 一種用於管理一可攜式計算裝置中之平行資源請求之方法，其包含：

自一第一用戶端產生一第一請求，該第一請求係在一第一執行線緒之內容脈絡中發出的；

將該第一請求轉遞至一資源；

由該資源來應答該第一請求且起始非同步處理；

在該資源中處理該第一請求同時允許該第一用戶端繼續該第一執行線緒中之處理；

由該資源來傳信該第一請求之該處理的完成；

在該資源中接收一第二請求，該第二請求導致該第一請求之該處理的完成，該第一請求之該處理的該完成包括將該資源之一本端代表更新至一新狀態及調用任何登記之回叫，該資源變得可用以服務於該第二請求；及

在該資源中處理該第二請求。

2. 如請求項1之方法，其中該第二請求源於該第一用戶端中。
3. 如請求項1之方法，其中該第二請求源於一第二用戶端中。
4. 如請求項1之方法，其中該第一請求經由一資源代理而被轉遞至該資源。
5. 如請求項1之方法，其中應答該第一請求進一步包含在該資源處理該第一請求之前允許該第一用戶端繼續該第一執行線緒中之處理。

6. 如請求項1之方法，其中由該資源來傳信該第一請求之該處理的完成使該資源成為可加入的。
7. 如請求項1之方法，其中該資源之該本端代表包含一資源代理及一執行構架中之任一者。
8. 如請求項1之方法，其中由該資源來應答該第一請求及起始非同步處理將該資源置於一不相干狀態，且該第二請求導致該資源返回至一相干狀態且可用以處理該第二請求。
9. 如請求項1之方法，其中在不接收該第二請求的情況下該資源變成可用。
10. 如請求項1之方法，其中該第二請求係在該第一執行線緒之內容脈絡中發出的。
11. 如請求項1之方法，其中該第二請求係在一第二執行線緒之內容脈絡中發出的。
12. 一種用於管理一可攜式計算裝置中之平行資源請求之電腦系統，該系統包含：
 - 一處理器，其可操作以用於：
 - 自一第一用戶端產生一第一請求，該第一請求係在一第一執行線緒之內容脈絡中發出的；
 - 將該第一請求轉遞至一資源；
 - 由該資源來應答該第一請求且起始非同步處理；
 - 在該資源中處理該第一請求同時允許該第一用戶端繼續該第一執行線緒中之處理；
 - 由該資源來傳信該第一請求之該處理的完成；

在該資源中接收一第二請求，該第二請求導致該第一請求之該處理的完成，該第一請求之該處理的該完成包括將該資源之一本端代表更新至一新狀態及調用任何登記之回叫，該資源變得可用以服務於該第二請求；及

在該資源中處理該第二請求。

13. 如請求項12之系統，其中該第二請求源於該第一用戶端中。
14. 如請求項12之系統，其中該第二請求源於一第二用戶端中。
15. 如請求項12之系統，其中該第一請求經由一資源代理而被轉遞至該資源。
16. 如請求項12之系統，其中應答該第一請求進一步包含在該資源處理該第一請求之前允許該第一用戶端繼續該第一執行線緒中之處理。
17. 如請求項12之系統，其中由該資源來傳信該第一請求之該處理的完成使該資源成為可加入的。
18. 如請求項12之系統，其中該資源之該本端代表包含一資源代理及一執行構架中之任一者。
19. 如請求項12之系統，其中由該資源來應答該第一請求及起始非同步處理將該資源置於一不相干狀態，且該第二請求導致該資源返回至一相干狀態且可用以處理該第二請求。
20. 如請求項12之系統，其中在不接收該第二請求的情況下該資源變成可用。

21. 如請求項12之系統，其中該第二請求係在該第一執行線緒之內容脈絡中發出的。

22. 如請求項12之系統，其中該第二請求係在一第二執行線緒之內容脈絡中發出的。

23. 一種用於管理一可攜式計算裝置中之平行資源請求之電腦系統，該系統包含：

用於自一第一用戶端產生一第一請求的構件，該第一請求係在一第一執行線緒之內容脈絡中發出的；

用於將該第一請求轉遞至一資源的構件；

用於由該資源來應答該第一請求且起始非同步處理的構件；

用於在該資源中處理該第一請求同時允許該第一用戶端繼續該第一執行線緒中之處理的構件；

用於由該資源來傳信該第一請求之該處理之完成的構件；

用於在該資源中接收一第二請求的構件，該第二請求導致該第一請求之該處理的完成，該第一請求之該處理的該完成包括將該資源之一本端代表更新至一新狀態及調用任何登記之回叫，該資源變得可用以服務於該第二請求；及

用於在該資源中處理該第二請求的構件。

24. 如請求項23之系統，其中該第二請求源於該第一用戶端中。

25. 如請求項23之系統，其中該第二請求源於一第二用戶端

中。

26. 如請求項23之系統，其進一步包含用於經由一資源代理將該第一請求轉遞至該資源的構件。
27. 如請求項23之系統，其中用於應答該第一請求的該構件進一步包含用於在該資源處理該第一請求之前允許該第一用戶端繼續該第一執行線緒中之處理的構件。
28. 如請求項23之系統，其中由該資源來傳信該第一請求之該處理的完成使該資源成為可加入的。
29. 如請求項23之系統，其中該資源之該本端代表包含一資源代理及一執行構架中之任一者。
30. 如請求項23之系統，其中用於由該資源來應答該第一請求及起始非同步處理的該構件將該資源置於一不相干狀態，且該第二請求導致該資源返回至一相干狀態且可用以處理該第二請求。
31. 如請求項23之系統，其中在不接收該第二請求的情況下該資源變成可用。
32. 如請求項23之系統，其中該第二請求係在該第一執行線緒之內容脈絡中發出的。
33. 如請求項23之系統，其中該第二請求係在一第二執行線緒之內容脈絡中發出的。
34. 一種包含一電腦可用媒體之電腦程式產品，該電腦可用媒體具有一體現於其中之電腦可讀程式碼，該電腦可讀程式碼經調適成被執行以實施一用於管理一可攜式計算裝置中之平行資源請求的方法，該方法包含：

自一第一用戶端產生一第一請求，該第一請求係在一第一執行線緒之內容脈絡中發出的；

將該第一請求轉遞至一資源；

由該資源來應答該第一請求且起始非同步處理；

在該資源中處理該第一請求同時允許該第一用戶端繼續該第一執行線緒中之處理；

由該資源來傳信該第一請求之該處理的完成；

在該資源中接收一第二請求，該第二請求導致該第一請求之該處理的完成，該第一請求之該處理的該完成包括將該資源之一本端代表更新至一新狀態及調用任何登記之回叫，該資源變得可用以服務於該第二請求；及

在該資源中處理該第二請求。

35. 如請求項34之電腦程式產品，其中該第二請求源於該第一用戶端中。

36. 如請求項34之電腦程式產品，其中該第二請求源於一第二用戶端中。

37. 如請求項34之電腦程式產品，其中該第一請求經由一資源代理而被轉遞至該資源。

38. 如請求項34之電腦程式產品，其中應答該第一請求進一步包含在該資源處理該第一請求之前允許該第一用戶端繼續該第一執行線緒中之處理。

39. 如請求項34之電腦程式產品，其中由該資源來傳信該第一請求之該處理的完成使該資源成為可加入的。

40. 如請求項34之電腦程式產品，其中該資源之該本端代表

包含一資源代理及一執行構架中之任一者。

41. 如請求項34之電腦程式產品，其中由該資源來應答該第一請求及起始非同步處理將該資源置於一不相干狀態，且該第二請求導致該資源返回至一相干狀態且可用以處理該第二請求。
42. 如請求項34之電腦程式產品，其中在不接收該第二請求的情況下該資源變成可用。
43. 如請求項34之電腦程式產品，其中該第二請求係在該第一執行線緒之內容脈絡中發出的。
44. 如請求項34之電腦程式產品，其中該第二請求係在一第二執行線緒之內容脈絡中發出的。

八、圖式：

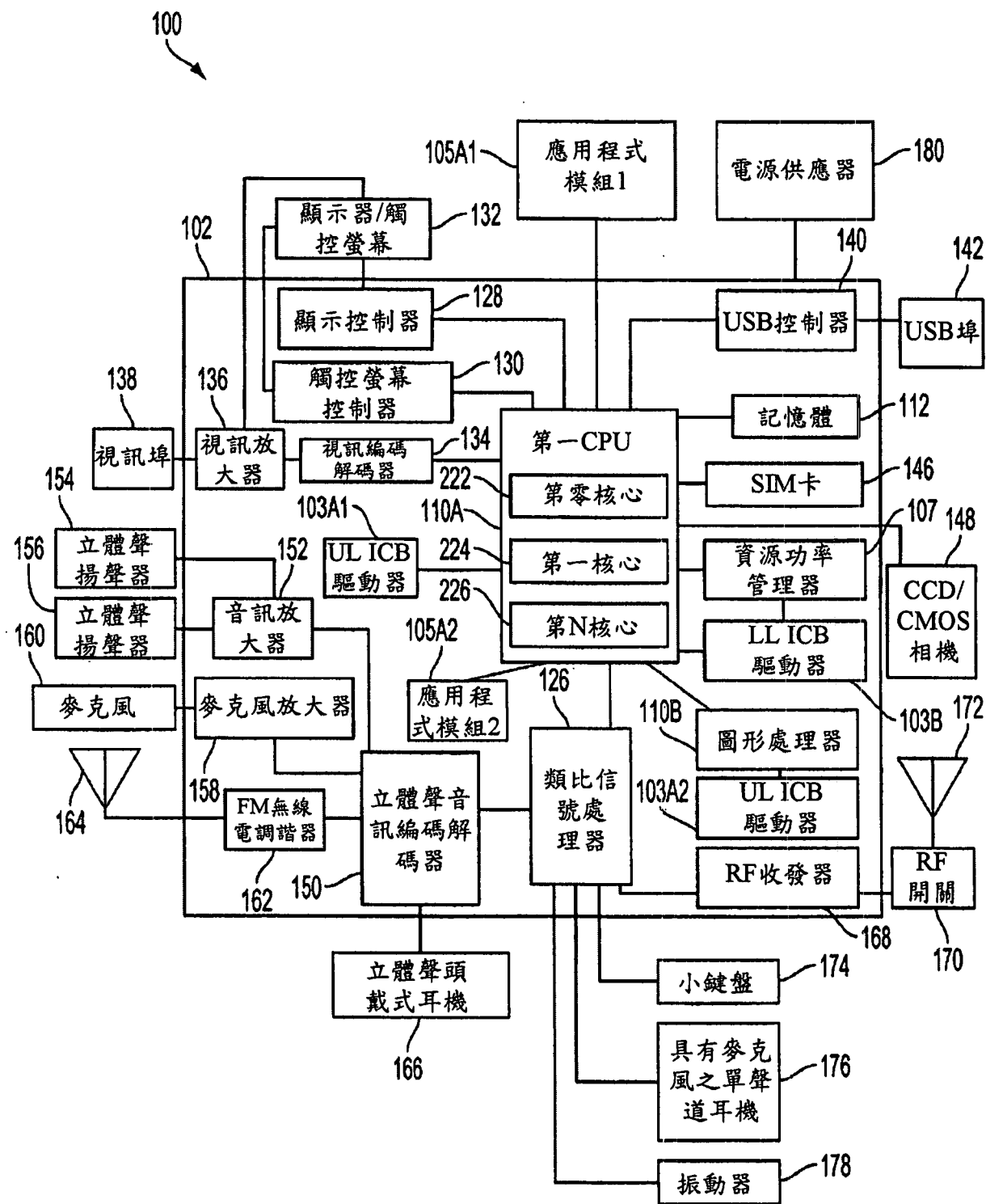


圖 1

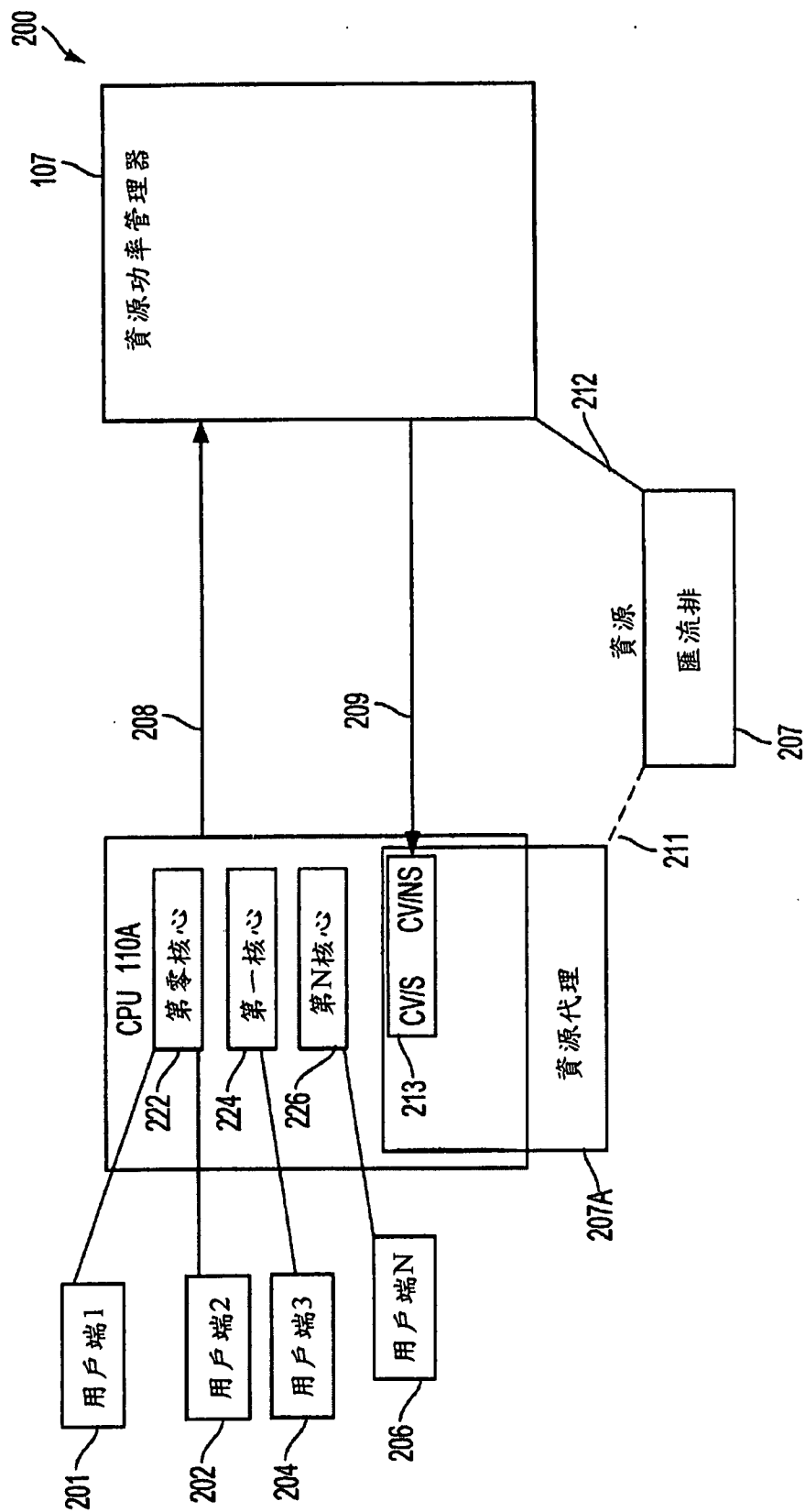


圖2

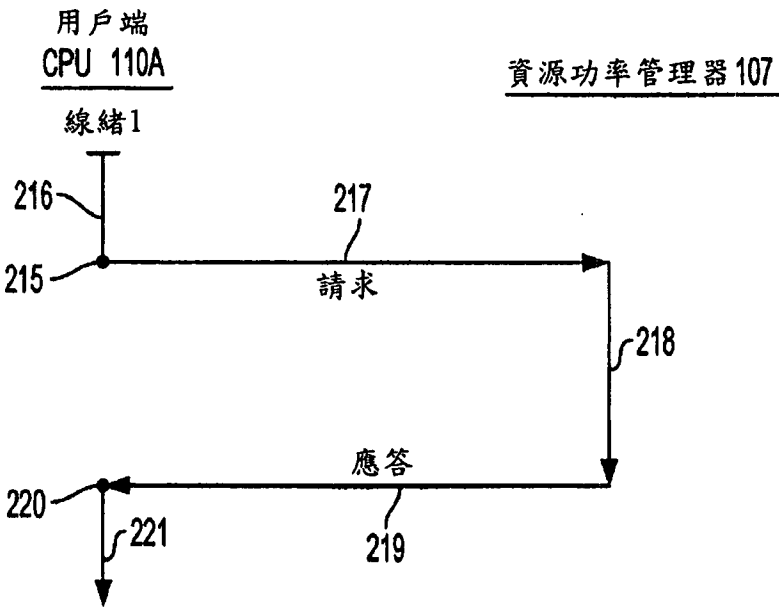


圖3

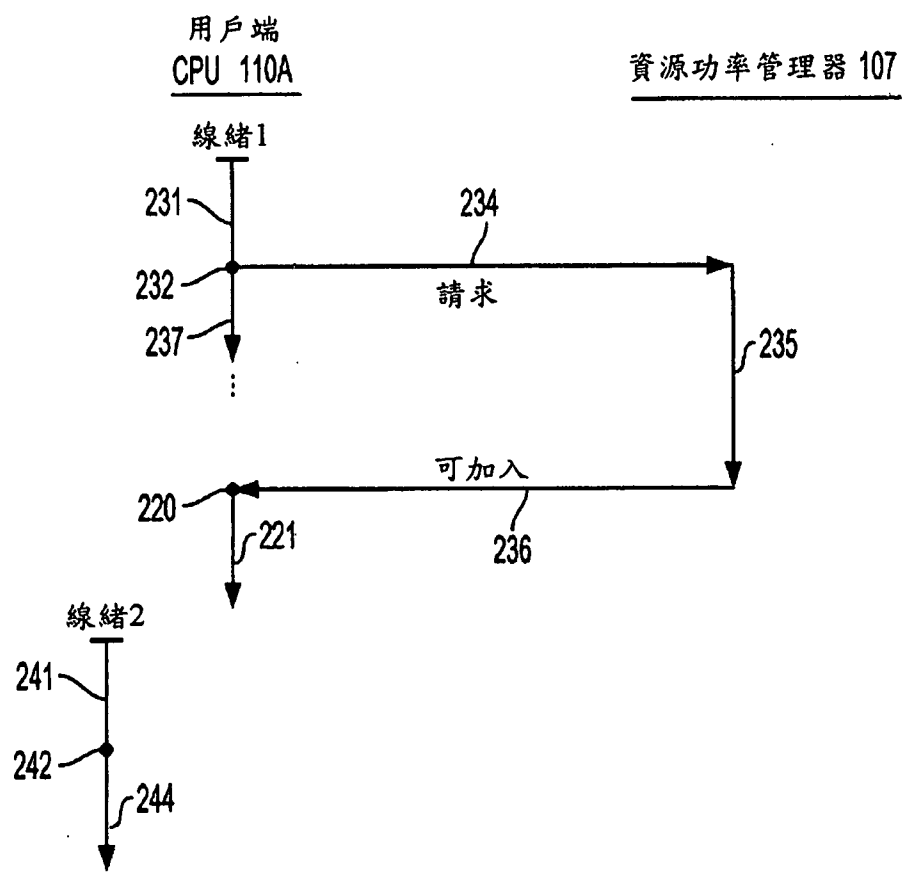


圖 4

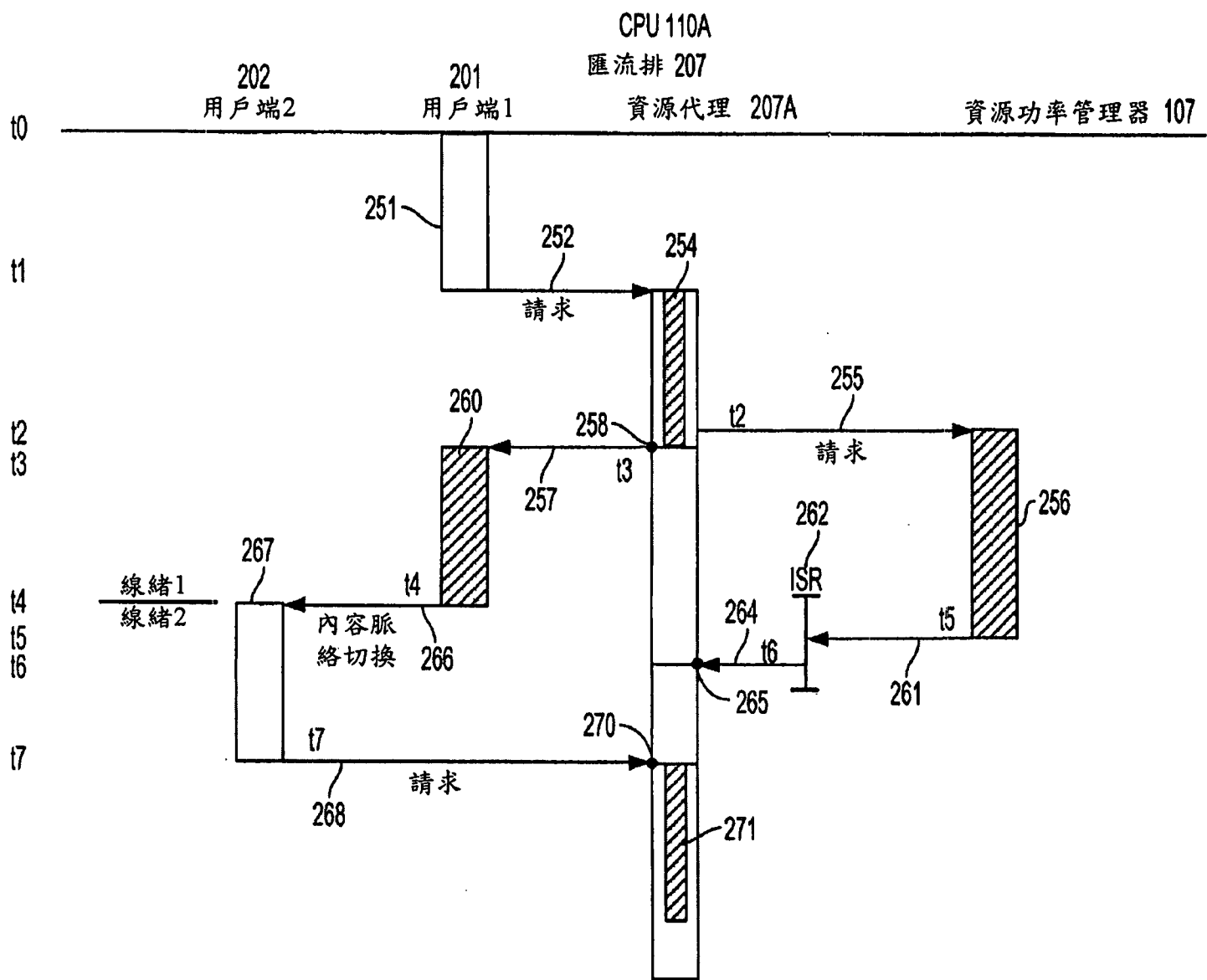
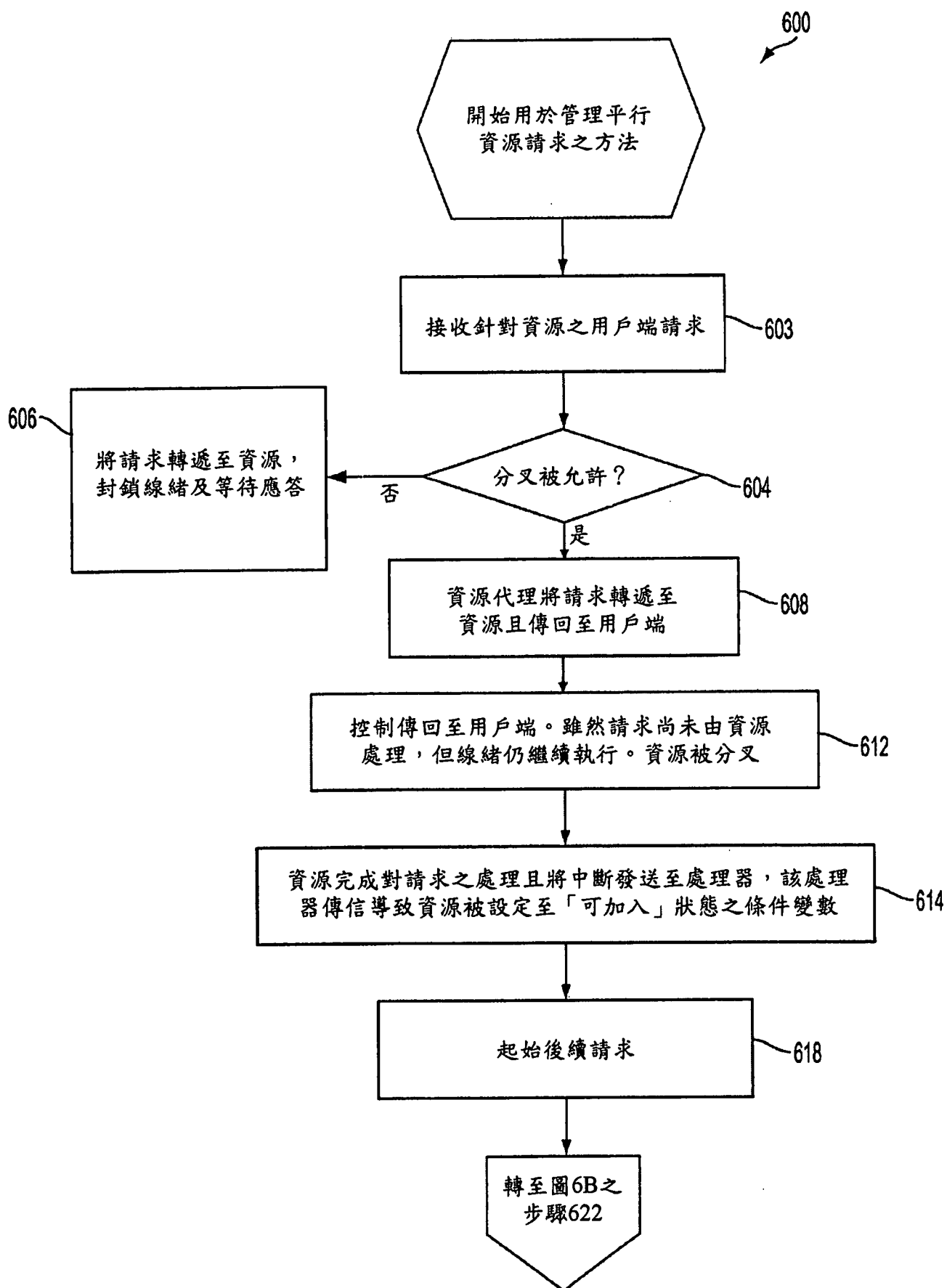


圖5



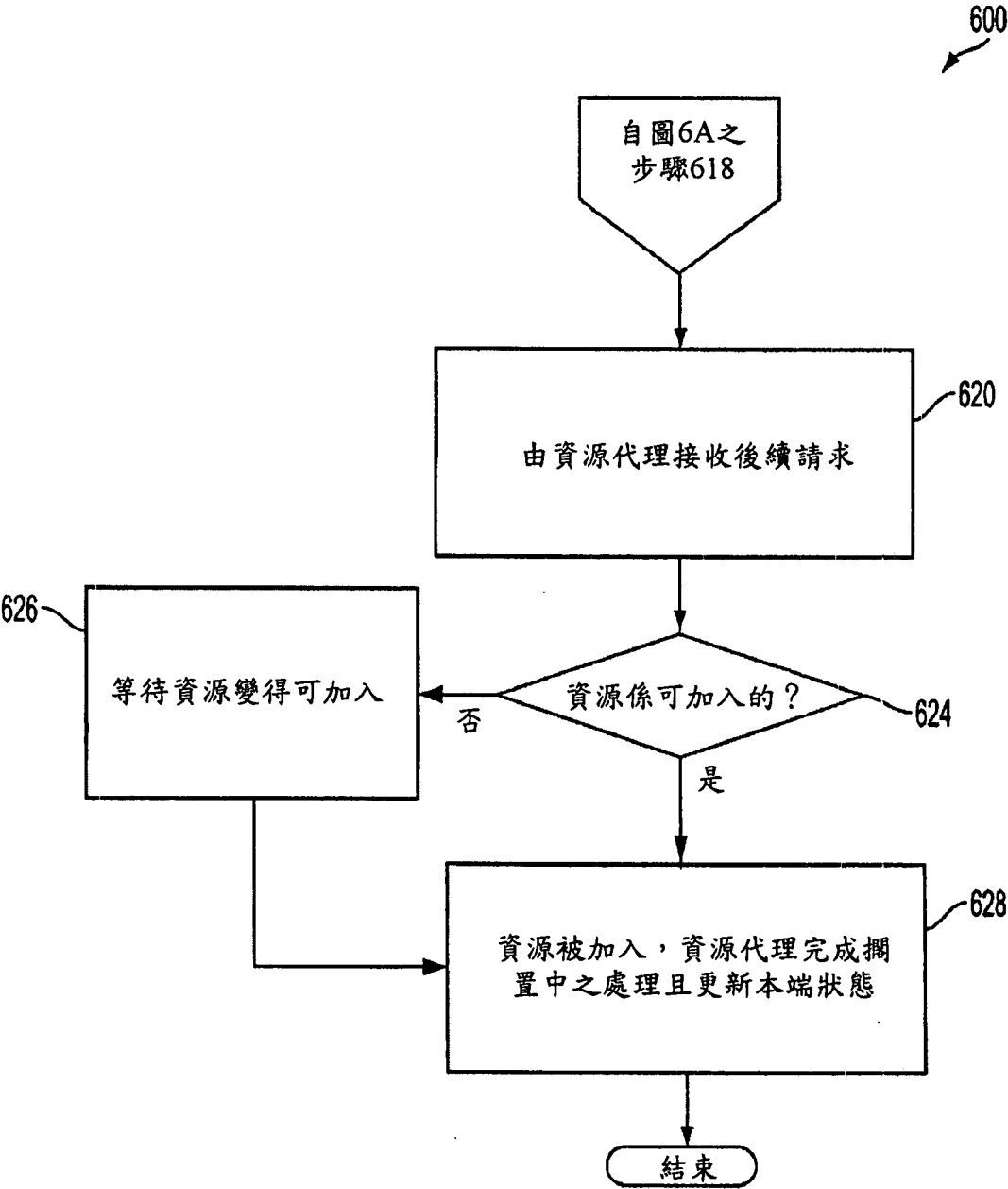


圖6B

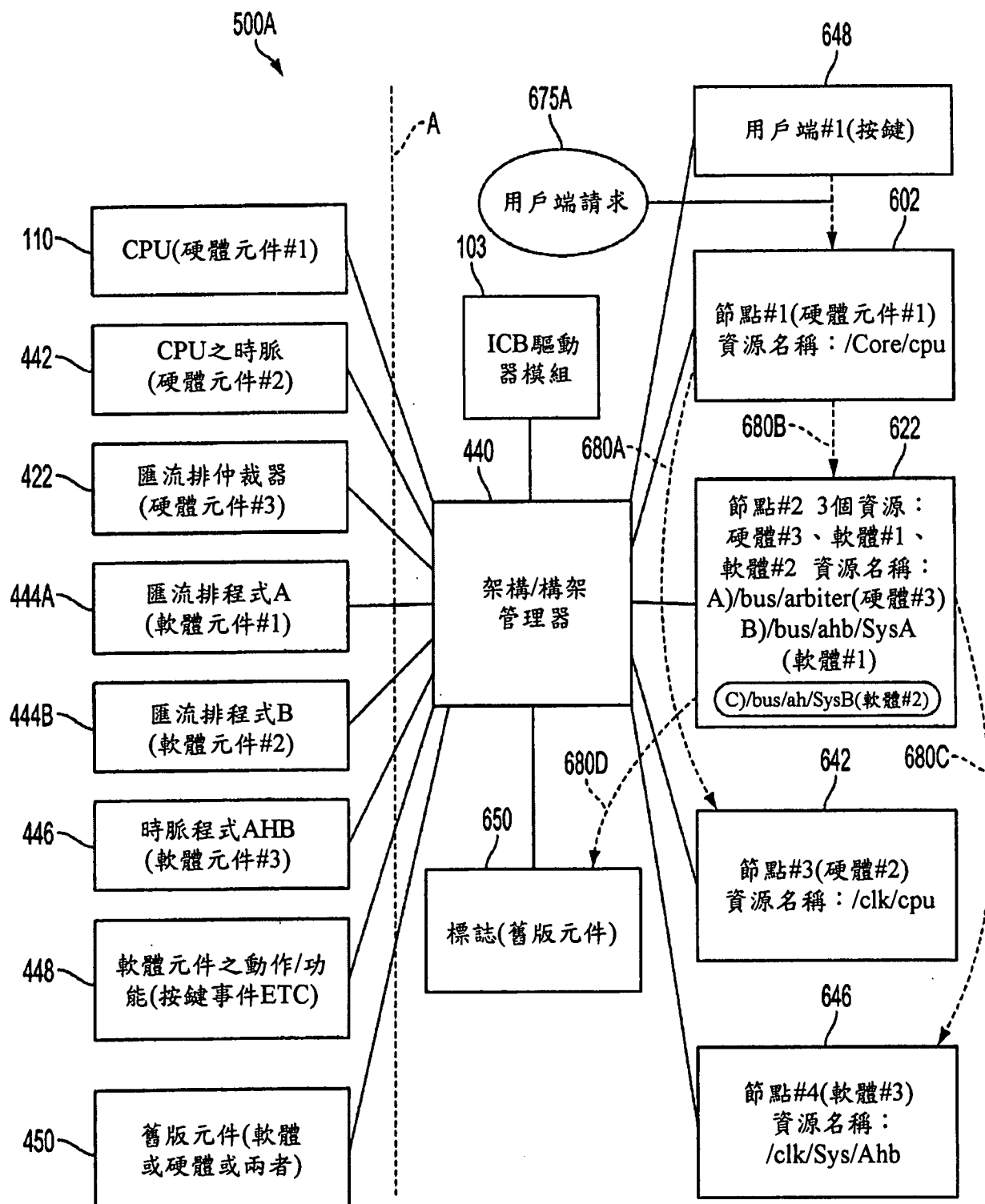


圖7A

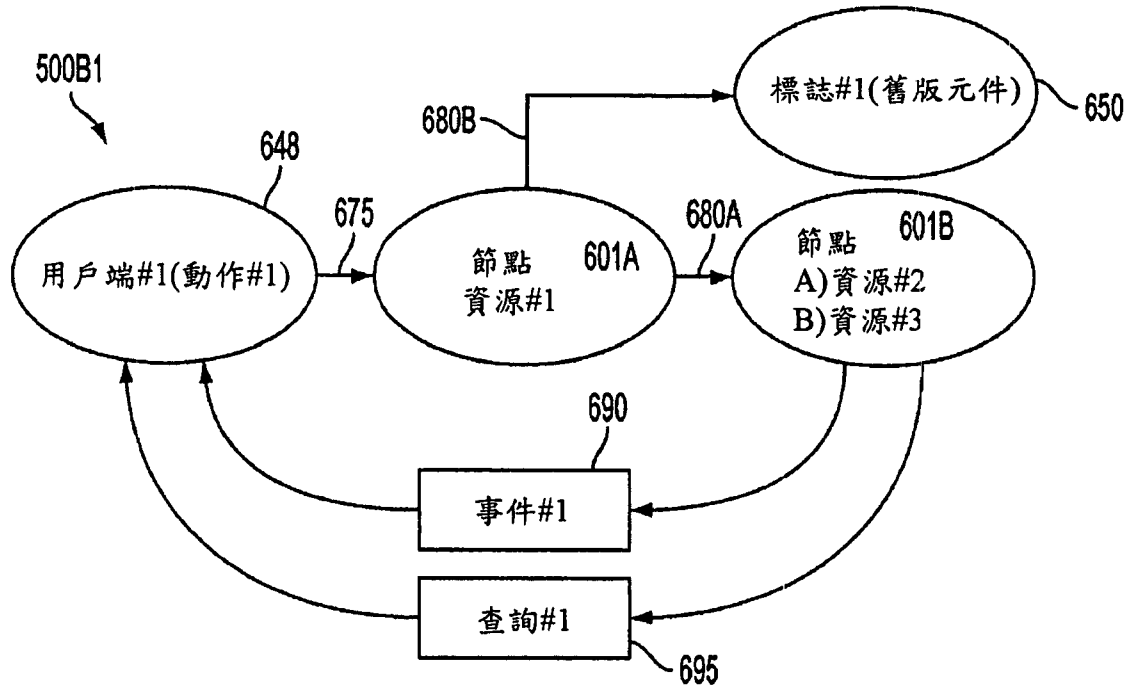


圖 7B

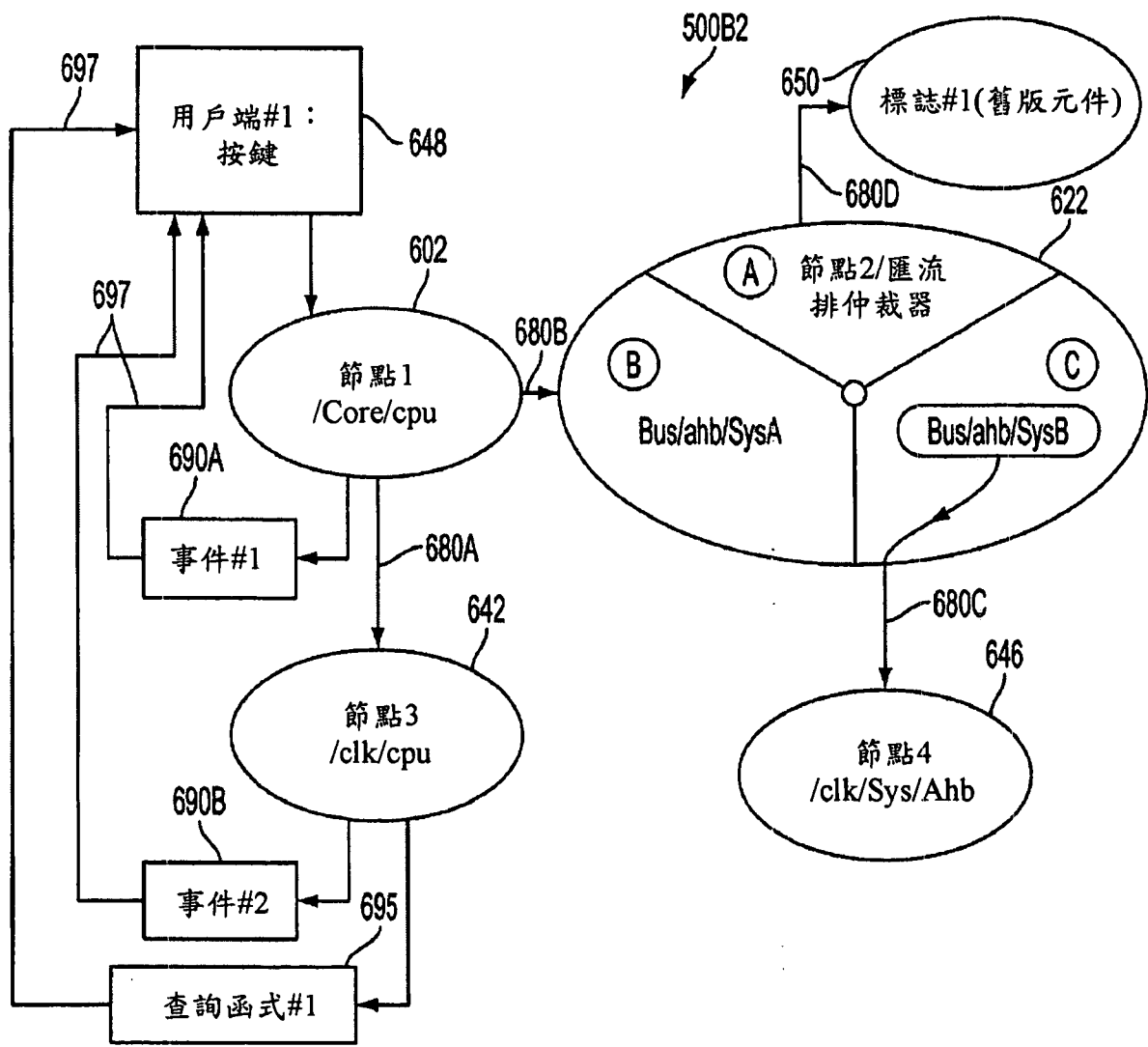


圖 7C

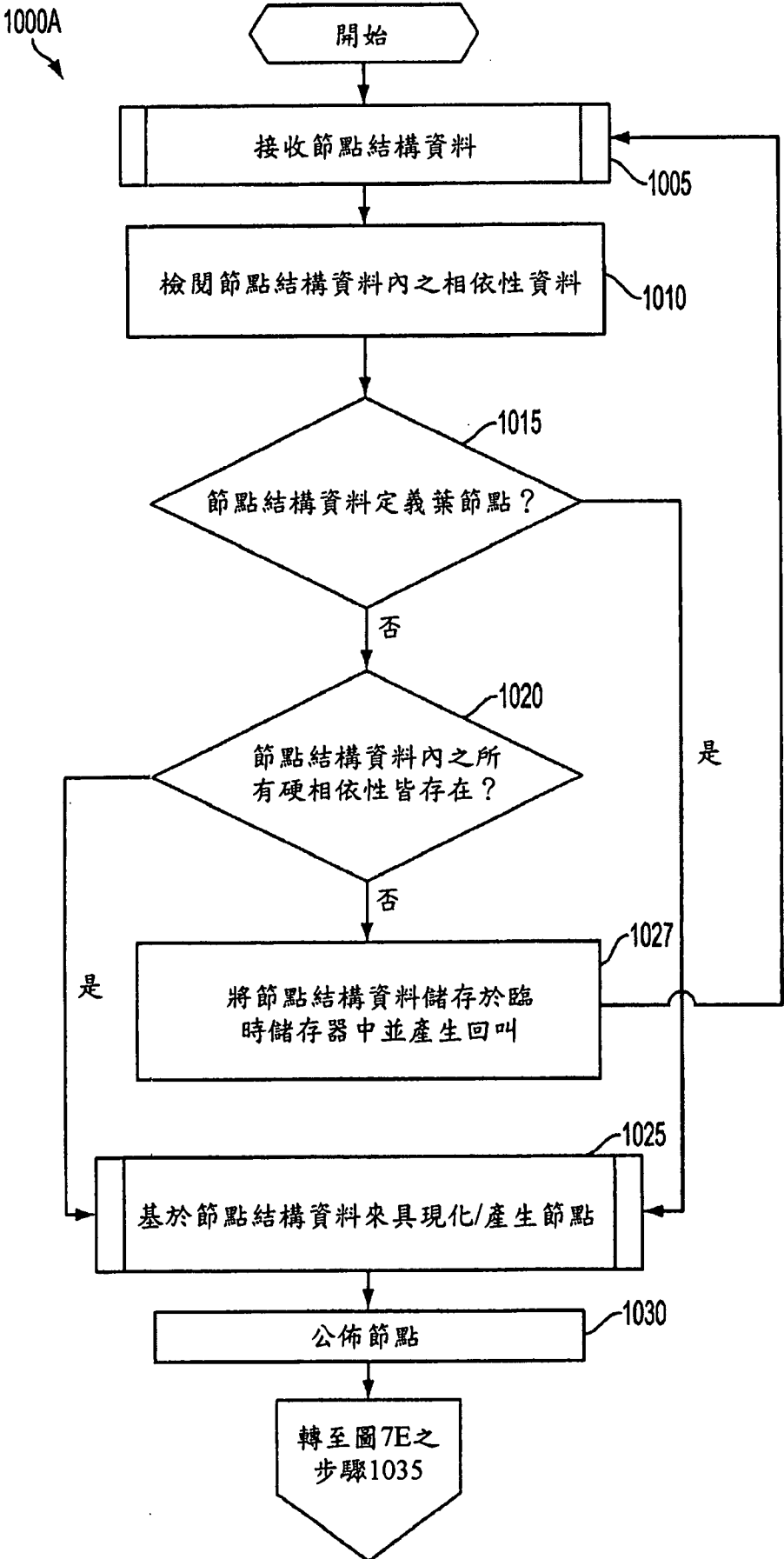


圖 7D

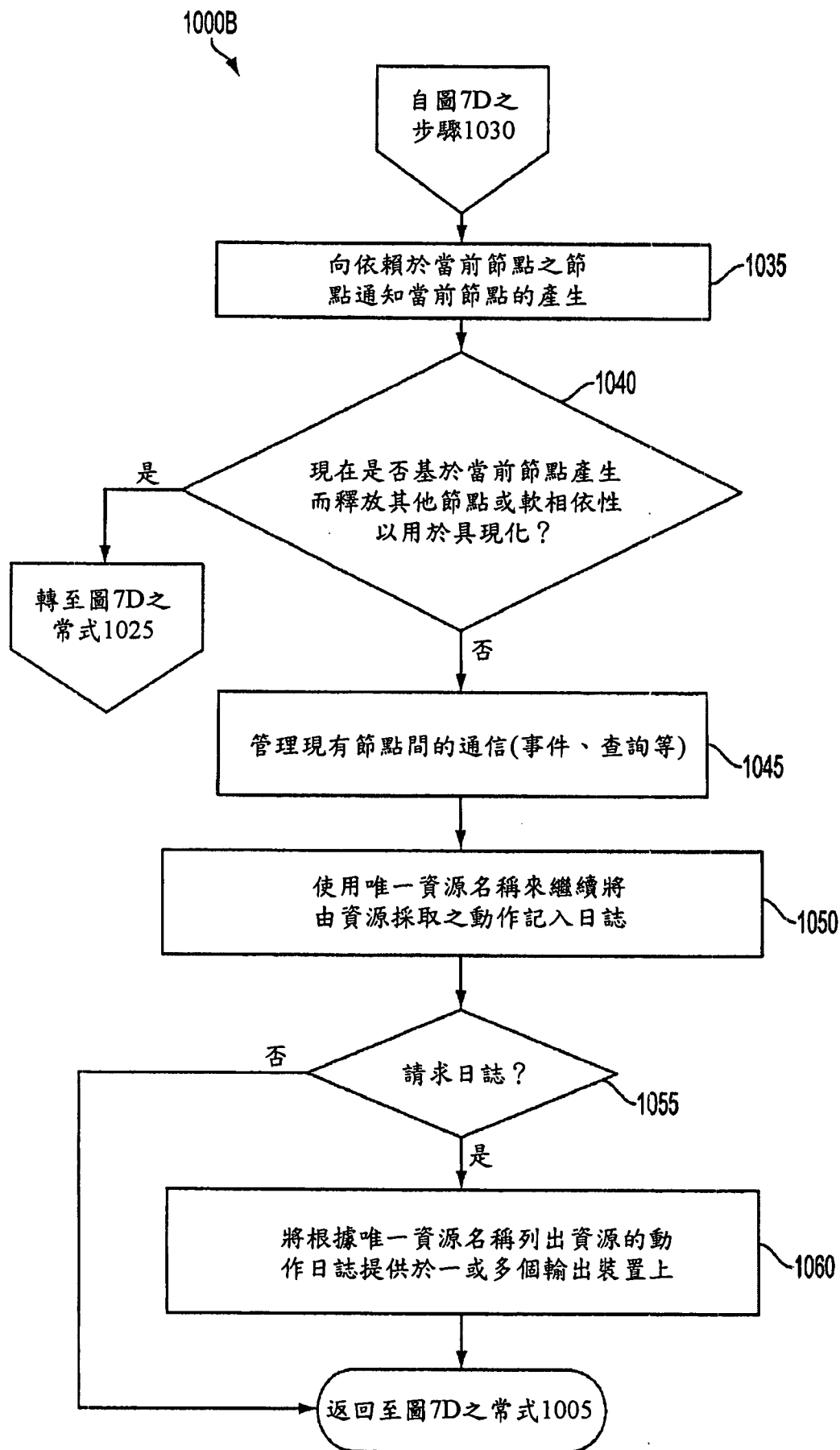


圖 7E

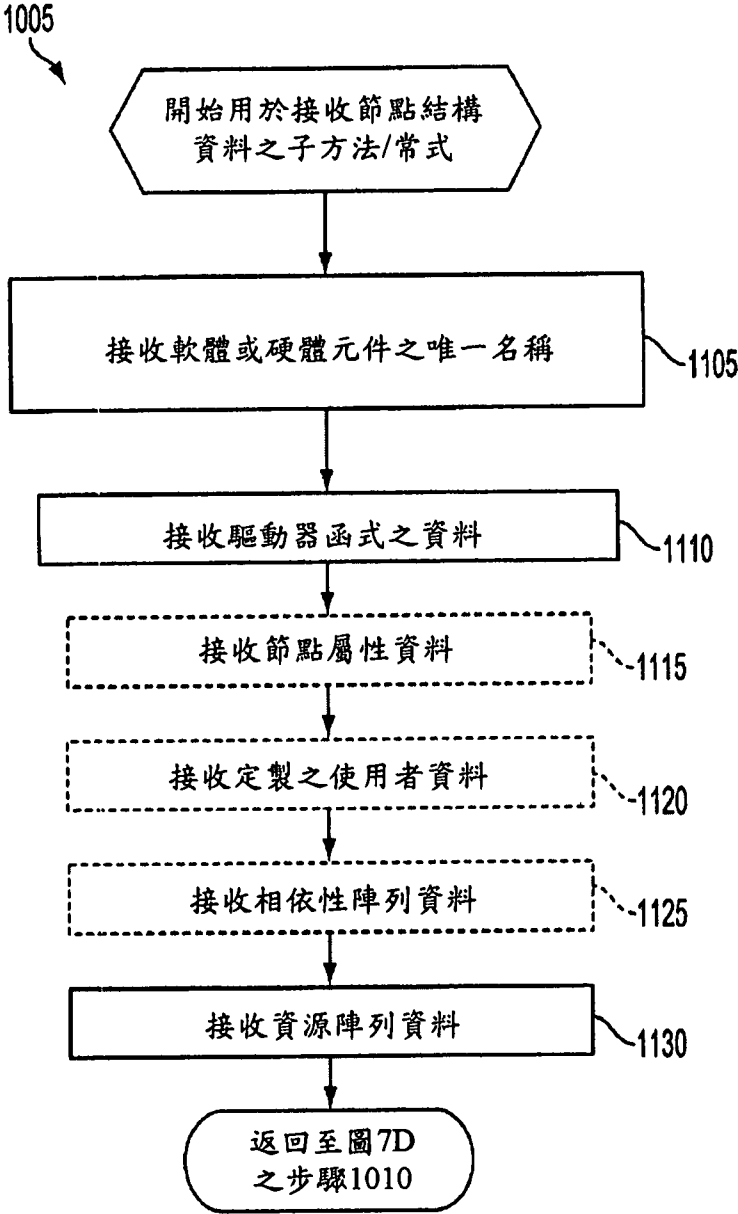


圖 8

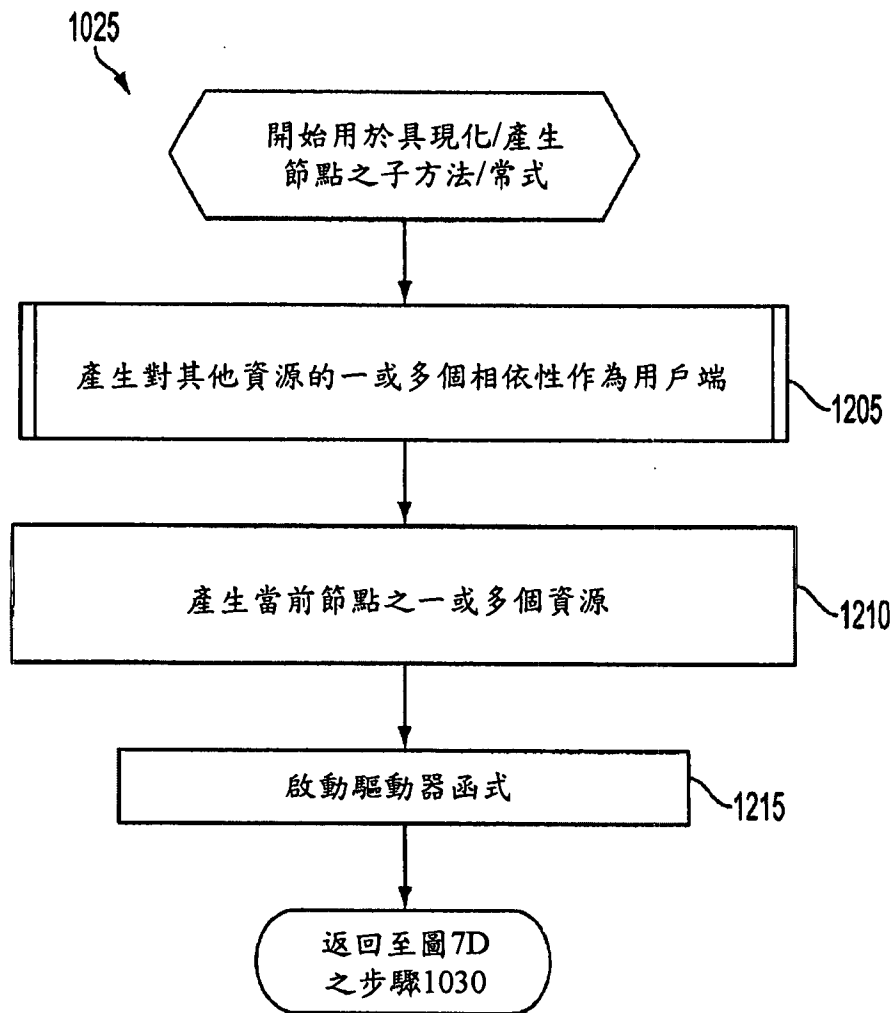


圖9

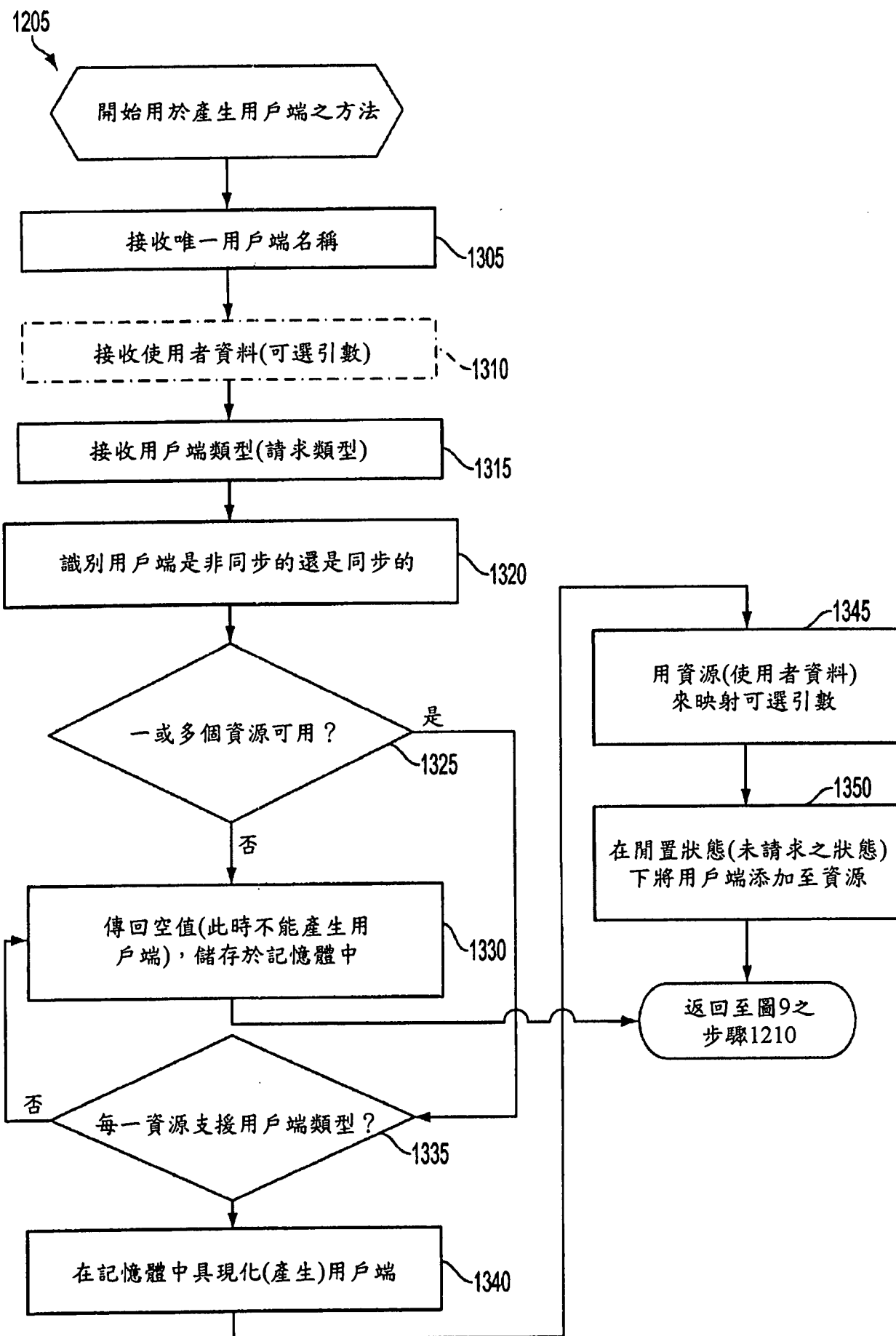


圖10

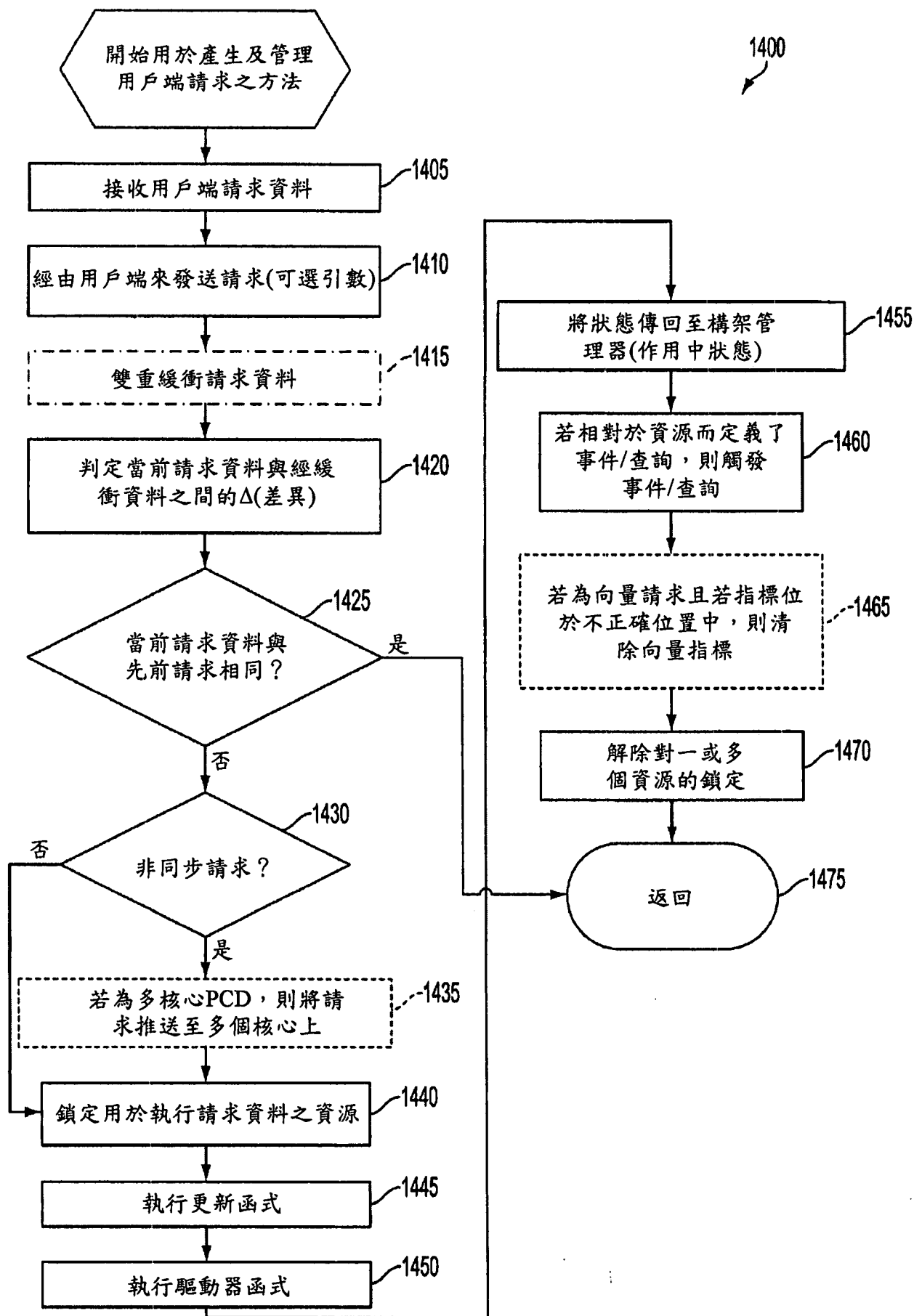


圖11