



(19)中華民國智慧財產局

(12)發明說明書公告本

(11)證書號數：TW I405070B1

(45)公告日：中華民國 102 (2013) 年 08 月 11 日

(21)申請案號：097150540 (22)申請日：中華民國 97 (2008) 年 12 月 24 日

(51)Int. Cl. : G06F1/04 (2006.01) G06F1/08 (2006.01)

(71)申請人：大同股份有限公司 (中華民國) TATUNG COMPANY (TW)

臺北市中山區中山北路 3 段 22 號

(72)發明人：黃奕翔 HUANG, I HSIANG (TW)

(74)代理人：詹銘文；蕭錫清

(56)參考文獻：

TW 200710630

TW 200802098

US 05812860

US 06425086

US 20040025069A1

Saowanee Saewong, "Power-Aware CPU Management in QoS-Guaranteed Systems." , 2007/09, PhD thesis, Electrical and Computer Engineering, Carnegie Mellon University

Bren Mochocki, Xiaobo Sharon Hu, Gang Quan, "Practical on-line DVS scheduling for fixed-priority real-time systems", 2005/03/10, the 11th IEEE Real Time and Embedded Technology and Applications Symposium

Aydin, H., Melhem, R., Mosse, D., Mejia-Alvarez, P., "Power-aware scheduling for periodic real-time tasks", 2004/05, IEEE Transactions on Computers, On page(s): 584 - 600 Volume: 53, Issue: 5

Gang Zeng, Tomiyama, H., Takada, H., Ishihara, T., "A Generalized Framework for System-Wide Energy Savings in Hard Real-Time Embedded Systems", 2008/12/20, IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, EUC '08. Volume 1, On page (s): 206 - 213

Chuan-Yue Yang, Jian-Jia Chen, Chia-Mei Hung, Tei-Wei Kuo, "System-Level Energy-Efficiency for Real-Time Tasks", 2007/05/09, 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing, ISORC '07., On page(s): 266 - 273

Agrawal, S., Yadav, R.S., Ranvijay, "System level energy aware fault tolerance approach for real time system", 2008/11/21, TENCON 2008 - 2008 IEEE Region 10 Conference, On page(s): 1 - 6

Dakai Zhu, Xuan Qi, Aydin, H., "Energy Management for Periodic Real-Time Tasks with Variable Assurance Requirements", 2008/08/27, 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications. RTCSA '08., On page(s): 259 - 268

審查人員：栗永欣

申請專利範圍項數：12 項 圖式數：8 共 0 頁

---

(54)名稱

頻率調整方法及使用此方法的電子裝置、電腦程式產品與電腦可讀取儲存媒體

METHOD FOR ADJUSTING FREQUENCY AND ELECTRONIC APPARATUS AND COMPUTER PROGRAM PRODUCT AND COMPUTER ACCESSIBLE STORAGE MEDIA TO STORE PROGRAM USING THE METHOD

(57)摘要

一種頻率調整方法及使用此方法的電子裝置、電腦程式產品與電腦可讀取儲存媒體。此頻率調整方法包括下列步驟。首先，取得在即時系統之工作排程中多個工作所對應的工作資訊，其中這些工作各自使用對應的工作時脈。接著，依據工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算每一工作各自的最小時脈調整比值。然後，依據這些工作的最小時脈調整比值以及其的優先權順序，計算出每一工作各自的理想頻率。最後，依據這些工作的理想頻率，調整這些工作所對應的工作時脈。

A method for adjusting frequency, an electronic apparatus, a computer program product, and a computer accessible storage media to store program using the method are provided. The method comprises the following steps. Firstly, a plurality of task information in accordance with a plurality of tasks in scheduling of real-time system is obtained, and each task is assigned its own clock frequency. Next, according to priority order from lowest to highest, a lowest clock adjustment ratio of every task is calculated by using corresponding task information respectively. Next, an ideal frequency of every task is calculated according to their lowest clock adjustment ratio and the priority order of the task respectively. Finally, the task clock of every task is adjusted according to these ideal frequencies.

S301~S304 . . . 本發明實施例的頻率調整方法的步驟

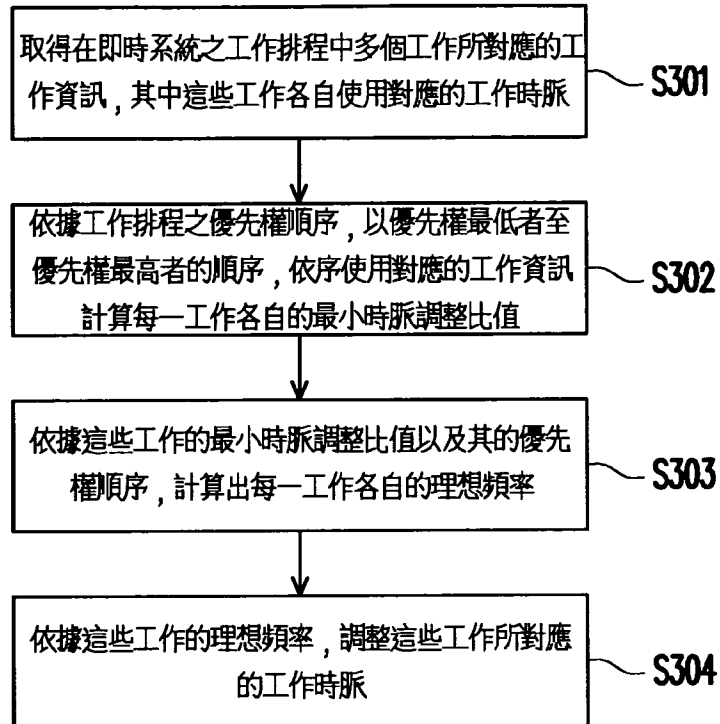


圖 3A

# 發明專利說明書

(本說明書格式、順序，請勿任意更動，※記號部分請勿填寫)

※申請案號： 97150540

※申請日： 97.11.14 ※IPC 分類：G06F 1/04 (2006.01)  
G06F 1/08 (2006.01)

## 一、發明名稱：

頻率調整方法及使用此方法的電子裝置、電腦程式產品與  
電腦可讀取儲存媒體 / METHOD FOR ADJUSTING  
FREQUENCY AND ELECTRONIC APPARATUS AND  
COMPUTER PROGRAM PRODUCT AND COMPUTER  
ACCESSIBLE STORAGE MEDIA TO STORE PROGRAM  
USING THE METHOD

## 二、中文發明摘要：

一種頻率調整方法及使用此方法的電子裝置、電腦程式產品與電腦可讀取儲存媒體。此頻率調整方法包括下列步驟。首先，取得在即時系統之工作排程中多個工作所對應的工作資訊，其中這些工作各自使用對應的工作時脈。接著，依據工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算每一工作各自的最小時脈調整比值。然後，依據這些工作的最小時脈調整比值以及其的優先權順序，計算出每一工作各自的理想頻率。最後，依據這些工作的理想頻率，調整這些工作所對應的工作時脈。

### 三、英文發明摘要：

A method for adjusting frequency, an electronic apparatus, a computer program product, and a computer accessible storage media to store program using the method are provided. The method comprises the following steps. Firstly, a plurality of task information in accordance with a plurality of tasks in scheduling of real-time system is obtained, and each task is assigned its own clock frequency. Next, according to priority order from lowest to highest, a lowest clock adjustment ratio of every task is calculated by using corresponding task information respectively. Next, an ideal frequency of every task is calculated according to their lowest clock adjustment ratio and the priority order of the task respectively. Finally, the task clock of every task is adjusted according to these ideal frequencies.

### 四、指定代表圖：

(一) 本案之指定代表圖：圖 3A

(二) 本代表圖之元件符號簡單說明：

S301~S304：本發明實施例的頻率調整方法的步驟

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：

### 【發明所屬之技術領域】

本發明是有關於一種頻率調整方法，且特別是有關於一種即時系統之工作排程的頻率調整方法及使用此方法的電子裝置、電腦程式產品與電腦可讀取儲存媒體。

### 【先前技術】

一般而言，中央處理器（Central Processing Unit, CPU）的工作排程（Scheduling）是多元程式規劃（Multi-task）作業系統的基礎。藉由中央處理器在不同工作（Task）之間的切換，作業系統可以讓電腦的輸出量提高。也就是說，藉由隨時保有一個工作在執行，以提高中央處理器的使用率。另外，在單一處理器系統裡，一次只能執行一個工作，無法多個工作同時執行，因此如果在多個工作的狀態下，其他的工作都必須等待中央處理器處理完當下的工作後再進行處理。

在下一個工作執行之前，若中央處理器處於閒置結束狀態，可藉由多元程式規劃，以有效地利用中央處理器的閒置結束時間。而多元程式規劃是將多個工作同時存放在記憶單元之中，當某個工作在等待的時候，作業系統就把中央處理器的使用權取走並且交給另一個工作。依上述方式進行，每當一個工作必須要等待的時候，中央處理器就可以先行處理另外一個工作。

此外，即時系統（Real-time System）有著較嚴格的時間限制。對於硬即時系統（hard real-time system）而言，所有工作更是必須要在所訂定的時間限制內完成，否則將會產生嚴重的後果（例如：核能發電廠的核子反應爐監視系統）。另外，

對於廣泛應用於今日生活且以電池為電源的可攜式嵌入式系統來說，需同時符合即時系統的期限要求以及低耗電量兩項要求，以便於增加可攜式嵌入式系統的使用時間，並避免產生嚴重的後果。因此，如何在最佳的工作排程下，達到降低中央處理器的使用頻率且符合即時系統對時序的要求，進而使得即時系統較為省電，將是一個重要的課題。

而最佳工作排程的取得方式之一，可利用在 "Practical voltage-scaling for fixed-priority RT-systems," Saowanee Saewong; Rajkumar, R. On Real-Time and Embedded Technology and Applications Symposium, 2003. Proceedings. The 9th IEEE Volume, Issue, 27-30 May 2003, pp. 106-114. 論文中所提出的 PM-clock 演算法 (Priority Monotonic Clock Frequency Assignment) 來實現。在 PM-clock 演算法中，會依據優先權順序由最高至最低的順序計算出每一工作的最小時脈調整比值，再依據這些最小時脈調整比值中計算出每一工作的理想頻率，最後根據這些理想頻率調整每一工作的工作頻率。其中這些工作的最小時脈調整比值的計算是透過函式 Energy-Minimizing-Freq01 來完成。

圖 1 為傳統實施例的函式 Energy-Minimizing-Freq01 的流程圖。請參照圖 1，在本實施例中，PM-clock 演算法在計算每一個工作  $\tau_i$  皆會呼叫一次函式 Energy-Minimizing-Freq01 來得到每一工作  $\tau_i$  的最小時脈調整比值，其中每一工作  $\tau_i$  對應的工作資訊皆會包括工作執行時間  $C_i$ 、工作觸發週期  $T_i$  及完成工作時限  $D_i$ ，其中  $i$  為一正整數。在步驟 S110 中，會設定所有參數的初始值，而參數  $\omega$  會設定為工作執行時間  $C_i$  除以頻率最大值  $f_{\max}$  (亦即此裝置的最大運作頻率)。

接著，步驟 S120 會判斷 IN\_BZP 的參數值是否為「TRUE (真)」。若步驟 S120 的判斷結果為「Yes」，亦即 IN\_BZP 的參數值為「TRUE」，則執行步驟 S130；反之，則執行步驟 140。而步驟 S130 會計算此工作  $\tau_i$  以頻率最大值  $f_{\max}$  執行時，其完成工作時限  $D_i$  內閒置時間的起始點，並存入參數  $\omega$  中，以及將 IN\_BZP 的參數值設定為「FALSE (偽)」，其中步驟 S131 用以計算並驗證此閒置時間的起始點是否正確，函數 hp 為優先權順序高於等於工作  $\tau_i$  的工作的集合，函數 floor 為取得小於括號中算式計算結果的最大整數。

接著，步驟 S150 會判斷參數  $\omega$  是否小於工作  $\tau_i$  的工作時限  $D_i$ ，若參數  $\omega$  小於工作  $\tau_i$  的工作時限  $D_i$  則回到步驟 S120；反之，則回傳  $\alpha$  的參數值，作為工作  $\tau_i$  的最小時脈調整比值。在參數  $\omega$  小於工作時限  $D_i$  時，會回到步驟 S120，並且由於 IN\_BZP 的參數值被設定為「FALSE」，所以此時步驟 S120 後會接著執行步驟 S140，以計算此工作  $\tau_i$  的時脈調整比值，並取其最小值存入參數  $\alpha$  中，其中函數 ceil 為取得大於括號中算式計算結果的最小整數。

而以函式 Energy-Minimizing-Freq01 來看，每一工作  $\tau_i$  皆要取得優先權順序較其高的工作之工作資訊，以計算出其最小時脈調整比值。然而，PM-clock 演算法是由優先權順序最高計算至最低，所以有很多相同的運算資訊會被重覆計算，並且計算的步驟通常會比較耗時，致使在計算每一工作的最小時脈調整比值會造成計算時間上的浪費。

## 【發明內容】



本發明提供一種頻率調整方法及使用此方法的電子裝置、電腦程式產品與電腦可讀取儲存媒體，藉此減少工作排程的運算步驟以降低即時系統排程的運算時間。

本發明提出一種頻率調整方法，其包括下列步驟。首先，取得在即時系統之工作排程中多個工作所對應的工作資訊，其中這些工作各自使用對應的工作時脈。接著，依據工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算每一工作各自的最小時脈調整比值，其中使用完整演算法去計算最低優先權者工作的最小時脈調整比值並記錄其中運算資訊，以及使用精簡演算法與前述運算資訊去計算非最低優先權者工作的最小時脈調整比值。然後，依據這些工作的最小時脈調整比值以及其的優先權順序，計算出每一工作各自的理想頻率。最後，依據這些工作的理想頻率，調整這些工作所對應的工作時脈。

在本發明之一實施例中，上述之工作所對應的工作資訊包括工作執行時間、工作觸發週期以及完成工作時限。

在本發明之一實施例中，上述之使用對應的工作資訊計算每一工作各自的最小時脈調整比值的步驟包括：利用完整演算法計算最低優先權順序的第一工作的最小時脈調整比值，並且記錄前述運算資訊中至少一閒置結束參數及其對應的鬆弛參數，以及依據所述閒置結束參數及其對應的鬆弛參數，利用精簡演算法計算這些工作中第二工作的最小時脈調整比值，其中所述閒置結束參數為第一工作的完成工作時限內至少一閒置時間的結束時間點，而所述鬆弛參數為所述閒置時間的時間長度，以及第二工作的優先權順序高於第一工作。

在本發明之一實施例中，上述之精簡演算法包括下列步驟：先設定最小時脈調整比值的初始值為 1，再依序取得所述閒置結束參數的其中之一及其對應的鬆弛參數，接著利用閒置結束參數及其對應的鬆弛參數依序計算出第二工作的時脈調整比值，若時脈調整比值小於最小時脈調整比值時，則記錄時脈調整比值作為第二工作的最小時脈調整比值，接著依據第二工作的工作執行時間調整鬆弛參數，以及在閒置結束參數小於第二工作的完成工作時限時，再次進行精簡演算法。

在本發明之一實施例中，上述之計算出每一工作的理想頻率的步驟包括：依據這些工作由優先權最高至優先權最低的順序，依序從優先權順序最高的工作進行計算，且每一工作的理想頻率是取優先權順序低於或等於其優先權順序之這些工作的最小時脈比值的最大值。

在本發明之一實施例中，頻率調整方法更包括在這些工作的其中之一理想頻率小於其前一工作的理想頻率時，依據優先權順序高於工作的這些工作的理想頻率，調整工作的理想頻率。

在本發明一實施例中，上述調整這些工作其中之一所對應的工作時脈的步驟包括下列步驟。當調降即時系統之處理器的工作電壓與即時系統之處理器的工作時脈時，先調降即時系統之處理器的工作時脈後，再調降即時系統之處理器的工作電壓。當調升即時系統之處理器的工作電壓與即時系統之處理器的工作時脈時，先調升即時系統之處理器的工作電壓後，再調即時系統之處理器的工作時脈。

本發明另提出一種電腦可讀取儲存媒體，用以儲存電腦程式。該電腦程式用以載入至電腦系統中並且使得該電腦系統執行上述頻率調整方法。

本發明亦提出一種電腦程式產品，當電子裝置載入該電腦程式並執行後，可完成上述頻率調整方法。

本發明再提出一種電子裝置，其包括處理器及記憶單元。處理器依據工作所對應的工作時脈而運行，其中處理器取得在即時系統之工作排程中多個工作所對應的工作資訊，且依據工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算每一工作各自的最小時脈調整比值，然後依據這些工作的最小時脈調整比值以及優先權順序，計算出每一工作各自的理想頻率，接著依據這些工作的理想頻率，調整這些工作所對應的工作時脈；其中，處理器使用完整演算法去計算最低優先權者工作的最小時脈調整比值並記錄其中運算資訊，以及使用精簡演算法與前述運算資訊去計算非最低優先權者工作的最小時脈調整比值。記憶單元耦接於處理器，儲存這些工作的工作資訊、最小時脈調整比值，以及理想頻率。

在本發明之一實施例中，上述之處理器利用完整演算法計算最低優先權順序的第一工作的最小時脈調整比值，並且記錄運算資訊中至少一閒置結束參數及其對應的鬆弛參數，以及依據所述閒置結束參數及對應的鬆弛參數，利用精簡演算法計算第二工作的最小時脈調整比值；其中，所述閒置結束參數為一閒置時間的結束時間，所述鬆弛參數為閒置時間的長度，且第二工作的優先權順序高於第一工作。

在本發明之一實施例中，上述之處理器更執行精簡演算法，以設定最小時脈調整比值的初始值為 1，接著依序取得所述閒置結束參數的其中之一及其對應的鬆弛參數以利用閒置結束參數及其對應的鬆弛參數依序計算出第二工作的時脈調整比值，在時脈調整比值小於最小時脈調整比值時，記錄時脈調整比值作為第二工作的最小時脈調整比值，以及依據第二工作的工作執行時間調整鬆弛參數，並且在閒置結束參數小於第二工作的完成工作時限時，更次執行精簡演算法。

基於上述，本發明的頻率調整方法、電子裝置、電腦程式產品及電腦可讀取儲存媒體，透過計算最低優先權工作的過程中，取得運算資訊中的閒置結束參數及對應鬆弛參數，再利用閒置結束參數及鬆弛參數計算出非最低優先權工作的最小時脈調整比值。藉此，可節省耗時的計算過程，以加速最佳排程的計算速度，減少最佳排程的計算時間。

為讓本發明之上述特徵和優點能更明顯易懂，下文特舉實施例，並配合所附圖式作詳細說明如下。

### 【實施方式】

圖 2 為根據發明一實施例之電子裝置的系統方塊圖。請先參照圖 2，電子裝置 200 包括處理器 210 及記憶單元 220。在本實施例中，電子裝置 200 可以為即時系統（Real-time System），而處理器 210 可以用來處理即時系統之工作排程（Scheduling），並且電子裝置 200 例如為可攜式電子裝置這一類可使用電池供應電源的電子裝置。記憶單元 220 耦接於處理器 210，用來儲存工作（Task）的工作資訊、最小時脈調整比值及理想頻率，其中工作資訊包括工作執行時間（Processor

cycle)、工作觸發週期 (Trigger Period)、完成工作時限 (Deadline)。前述工作執行時間是指不降低 CPU 運作頻率的情況下完成該工作所需執行時間；工作觸發週期是指發生該工作(或該工作進入工作排程)的時間週期；而完成工作時限是指從發生該工作(或該工作進入工作排程)的時間算起，完成該工作的相對時限。而工作執行時間、工作觸發週期以及完成工作時限，分別以  $C_i$ 、 $T_i$ 、 $D_i$  表示之，其中  $i$  為大於等於 1 的正整數。值得一提的是，記憶單元 220 可整合進處理器 210 中，亦即處理器 210 會包含記憶單元 220，以處理即時系統之工作排程。

圖 3A 為根據本發明一實施例之頻率調整方法的流程圖。請參照圖 2 及圖 3A，首先，在步驟 S301 中，處理器 210 會取得在即時系統之工作排程中多個工作所對應的工作資訊，並且處理器 210 會依據所處理的工作所對應的工作時脈而運行，以執行這些工作。在此為了方便說明，在本實施例中，以取得 3 個工作作為說明，且這 3 個工作分別以  $\tau_1$ 、 $\tau_2$ 、 $\tau_3$  表示之，以優先權順序來排列則為  $\tau_1 > \tau_2 > \tau_3$ 。其中，工作  $\tau_1$  對應的工作資訊  $C_1$ 、 $T_1$ 、 $D_1$  為  $\{7 * f_{\max}, 20, 20\}$ ；工作  $\tau_2$  對應的工作資訊  $C_2$ 、 $T_2$ 、 $D_2$  為  $\{5 * f_{\max}, 28, 28\}$ ；工作  $\tau_3$  對應的參數值  $C_3$ 、 $T_3$ 、 $D_3$  為  $\{3 * f_{\max}, 30, 30\}$ 。另外，上述  $f_{\max}$  為頻率最大值（亦即該系統的處理器的最大運作頻率），而上述對應參數  $C_i$  項的表示式  $C_i * f_{\max}$  係指以  $f_{\max}$  為頻率所需的執行時間為  $C_i$ ，並非指  $C_i$  乘以  $f_{\max}$ 。而工作  $\tau_1$ 、 $\tau_2$ 、 $\tau_3$  的排程會如圖 3B 所示，圖 3B 為圖 3A 之工作排程示意圖，其縱軸為相對於頻率最大值  $f_{\max}$  的倍數，其橫軸為時間  $t$ 。

接著，在步驟 S302 中，處理器 210 會依據工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算每一工作各自的最小時脈調整比值。其中，處理器 210 會使用完整演算法（稍後說明）與對應的工作資訊去計算最低優先權者工作的最小時脈調整比值並記錄其中運算資訊，以及使用精簡演算法（稍後說明）與前述運算資訊去計算這些工作中非最低優先權者的最小時脈調整比值。

換言之，處理器 210 會利用完整演算法先計算出工作  $\tau_3$  的最小時脈調整比值。請參照圖 3B，在計算過程中，處理器 210 會取得小於或等於工作  $\tau_3$  完成工作時限的閒置結束參數（亦即閒置時間的結束時間點），而從圖示中可以得知閒置結束參數分別為 20 及 28。由於工作  $\tau_3$  要在滿足工作  $\tau_1$  及  $\tau_2$  的執行需求後才獲得執行的權力，因此執行工作  $\tau_3$  時，尚須考慮工作  $\tau_1$  及工作  $\tau_2$ ，所以處理器 210 於閒置結束參數為 20 時，其工作量總共為 15（即工作  $\tau_1$ +工作  $\tau_2$ +工作  $\tau_3$ ），而其鬆弛參數（閒置時間的時間長度）為 5（即 20-15）。另考量工作  $\tau_3$  於閒置結束參數為 28 的執行工作時間與鬆弛參數，則處理器 210 執行工作的時間總共為 22（即工作  $\tau_1$ \*2+工作  $\tau_2$ +工作  $\tau_3$ ），而其鬆弛參數為 6（即 28-22）。因此，工作  $\tau_3$  的最小時脈調整比值會等於  $\min\{15/20, 22/28\}=0.75$ 。由於工作  $\tau_2$  執行於工作  $\tau_3$  之前，所以工作  $\tau_3$  的工作執行時間  $C_3$  對工作  $\tau_2$  而言會等同閒置時間，因此對於工作  $\tau_2$  而言只需將工作  $\tau_3$  的計算過程中的鬆弛參數進行修正，便可以獲得工作  $\tau_2$  的鬆弛參數。例如，工作  $\tau_2$  所對應的鬆弛參數會分別修正為 8（即 5+3）及 9（即 6+3）。此鬆弛參數的修正結果（即運算資訊）可以被記錄下來，以備後續運算工作  $\tau_1$  與  $\tau_2$  的最小

時脈調整比值的運算過程所利用。其中，上述工作  $\tau_3$  的計算過程則為完整演算法的運算過程。

接著計算工作  $\tau_2$ ，由於閒置結束參數 20 及 28 為小於或等於工作  $\tau_2$  的完成工作時限  $D_2$  (即 28)，以致這些閒置結束參數皆會在完成工作時限  $D_2$  內皆會出現，所以閒置結束參數 20 及 28 皆會被使用到。另外，工作  $\tau_2$  的鬆弛參數已經由前述完整演算法的運算過程中求得，亦即對應於閒置結束參數 20 及 28 的鬆弛參數分別為 8 及 9，因此可以進一步求得其執行的時間總共分別為 12(即 20-8)及 19(即 28-9)。因此，工作  $\tau_2$  的最小時脈調整比值會等於  $\min\{12/20, 19/28\}=0.60$ 。其中，上述工作  $\tau_2$  的計算過程因利用工作  $\tau_3$  計算過程中的運算資訊而不需要重新進行一次完整演算法，故上述工作  $\tau_2$  的計算過程稱之為精簡演算法。另外，在使用精簡演算法計算工作  $\tau_2$  的最小時脈調整比值的過程中，也可以順便計算工作  $\tau_1$  的鬆弛參數。同理可推，對於工作  $\tau_1$  而言，工作  $\tau_2$  的工作執行時間會等同閒置時間，因此可以將鬆弛參數分別修正為 13 (即 8+5) 及 14 (即 9+5)。

在計算工作  $\tau_1$  時，由於工作  $\tau_1$  的鬆弛參數已經由前述工作  $\tau_2$  的精簡演算法的運算過程中求得，因此亦可以再一次使用精簡演算法來計算工作  $\tau_1$  的最小時脈調整比值。因為閒置結束參數 28 大於工作  $\tau_1$  的完成工作時限  $D_1$  (即 20)，故在此只會使用到閒置結束參數 20，而其對應的鬆弛參數為 13，而執行的時間總共為 7(即 20-13)。因此，工作  $\tau_1$  的最小時脈調整比值會等於  $7/20=0.35$ 。

在計算出工作  $\tau_1 \sim \tau_3$  的最小時脈調整比值後，如步驟 S303 所示，處理器 210 會依據這些工作的最小時脈調整比值

以及其優先權順序，計算出工作  $\tau_1 \sim \tau_3$  各自的理想頻率，其中上述工作的理想頻率的計算方式例如為取優先權順序低於或等於其的這些工作的最小時脈頻率的<sub>最大值</sub>。換言之，工作  $\tau_1$  的理想頻率會等於  $\max\{0.35, 0.60, 0.75\}=0.75$ ，工作  $\tau_2$  的理想頻率會等於  $\max\{0.60, 0.75\}=0.75$ ，工作  $\tau_3$  的理想頻率會等於  $\max\{0.75\}=0.75$ 。最後，如步驟 S304 所示，依據工作  $\tau_1 \sim \tau_3$  各自的理想頻率，調整各自的工作時脈，使得工作  $\tau_1 \sim \tau_3$  的工作時脈的頻率皆降低為頻率最大值  $f_{\max}$  的 0.75 倍。藉此，可在符合每一工作的工作資訊下，降低處理器 210 工作  $\tau_1 \sim \tau_3$  各自的工作時脈的頻率，進而達到節能的作用。

此外，由上述頻率調整方法可再延伸出一實施例。圖 4 為根據本發明另一實施例的頻率調整方法的流程圖。請參照圖 2、圖 3A 及圖 4，在本實施例中，步驟 S401 及 S407 可參照圖 3A 相似部份所述，故在此不再贅述。在步驟 S402 中，處理器 210 會利用完整演算法計算最低優先權順序的第一工作的最小時脈調整比值，並且記錄前述運算資訊中至少一閒置結束參數及其對應的鬆弛參數。接著，步驟 S403 中，處理器 210 依據所述閒置結束參數及其對應的鬆弛參數，利用精簡演算法計算這些工作中第二工作的最小時脈調整比值，其中第二工作的優先權順序高於第一工作。然後在步驟 S404 中，處理器 210 會判斷工作排程中所有的第二工作的最小時脈調整比值是否被計算出，若判斷結果為「否」，代表還有第二工作的最小時脈調整比值未被計算出，則回到步驟 S403，以計算其他第二工作的最小時脈調整比值；若判斷結果為「是」，代表所有第二工作的最小時脈調整比值皆被計算出，則接著執行步驟 S405。



以上述三個工作  $\tau_1 \sim \tau_3$  的計算過程為例，步驟 S402 則對應至工作  $\tau_3$ （亦即第一工作）的計算過程。而步驟 S403 則對應至工作  $\tau_1$  及  $\tau_2$  的計算過程。由於排程中優先權順序高於工作  $\tau_3$  的有工作  $\tau_1$  及  $\tau_2$ （亦即第二工作），所以步驟 S404 第一次的判斷結果會為「否」，使得處理器 210 再次執行步驟 403；而步驟 S404 第二次的判斷結果會為「是」，則接著執行步驟 S405，結束計算工作  $\tau_1 \sim \tau_3$  的最小時脈調整比值。

在步驟 S405 中，處理器 210 會由工作  $\tau_1 \sim \tau_3$  中優先權最高至優先權最低的順序，依序從工作  $\tau_1$  開始進行計算，且每一工作的理想頻率是取優先權順序低於或等於其優先權順序之這些工作的最小時脈比值的最大值。此步驟 405 的計算過程可參照步驟 303 所述計算過程，在此不再贅述。在步驟 S406 中，處理器 210 會在工作  $\tau_2 \sim \tau_3$  的其中之一理想頻率小於其前一工作的理想頻率時，依據優先權順序高於此工作的這些工作的理想頻率，調整工作的理想頻率。在工作  $\tau_1 \sim \tau_3$  皆計算出其理想頻率後，接著執行步驟 S407。

舉例來說，假設工作  $\tau_3$  所算出的理想頻率為 0.7，會小於工作  $\tau_2$  的理想頻率（0.75），所以會對工作  $\tau_3$  的理想頻率進行調整。在此會先減去工作  $\tau_1$  及  $\tau_2$  利用理想頻率調整其工作時脈後所佔用的時間，以求得工作  $\tau_3$  的最大執行時間，亦即工作  $\tau_3$  的完成工作時限  $D_3$ （30）減去  $(C_1 * 2 + C_2) / 0.75 = 4.7$ 。接著，求出工作  $\tau_3$  以最大頻率執行所需時間與經調整頻率後最大執行時間的比值，亦即  $3 / 4.7 = 0.638$ ，此數值 0.638 即為工作  $\tau_3$  調整後的最小時脈調整比值，再利用調整後此最小時脈調整比值即可求出工作  $\tau_3$  調整後的理想頻率。由於工作  $\tau_3$  為最

低優先權的工作，所以其最小時脈調整比值（0.638）亦即理想頻率。

更者，可以對步驟 S403 中所述的精簡演算法作進一步的解說，以使本領域通常知識者可了解上述實施例。圖 5 為圖 4 步驟 S403 中精簡演算法的細部流程圖。請參照圖 4 及圖 5，在步驟 S403 中，首先會執行步驟 S501，先設定最小時脈調整比值的初始值為 1。接著步驟 S502 會依序取得所述閒置結束參數的其中之一及其對應的鬆弛參數，並且步驟 S503 會利用閒置結束參數及其對應的鬆弛參數依序計算出第二工作的時脈調整比值。

而步驟 S504 會判斷時脈調整比值是否小於最小時脈調整比值。若時脈調整比值小於最小時脈調整比值時，則執行步驟 S505 以記錄時脈調整比值作為第二工作的最小時脈調整比值；反之，則忽略步驟 S505 以直接執行步驟 S506。接著，步驟 S506 會依據第二工作的工作執行時間  $C_i$  調整所使用的鬆弛參數。如步驟 S507 所示，會判斷閒置結束參數是否小於第二工作的完成工作時限  $D_i$ 。在閒置結束參數小於第二工作的完成工作時限  $D_i$  時，會回到步驟 S502 以再次進行精簡演算法；反之，則往下執行步驟 S404。

以計算上述工作  $\tau_2$  的最小時脈調整比值為例，其中閒置結束參數及鬆弛參數請參照上述的說明。首先，會將最小時脈調整比值的初始值設定為 1，在第一次執行步驟 502 時，會取得的閒置結束參數為 20 及對應的鬆弛參數為 8，而工作  $\tau_2$  執行的時間總共為 12（即 20-8），藉此可得到時脈調整比值會為 12/20（即 0.6）。由於時脈調整比值（即 0.6）小於最小時脈調整比值（即 1），所以最小時脈調整比值會變更為 0.6，

接著會將鬆弛參數修正為 13 (即  $8+5$ )，以加入工作  $\tau_2$  的工作執行時間  $C_2$ 。由於此時所使用的閒置結束參數 (即 20) 小於工作  $\tau_2$  的完成工作時限  $D_2$  (即 28)，所以會再回到步驟 502。

在第二次執行步驟 502 時，會取得的閒置結束參數為 28 及對應的鬆弛參數為 9，而工作  $\tau_2$  執行的時間總共會為 19 (即  $28-9$ )，藉此可得到時脈調整比值會為  $19/28$  (即 0.679)。由於時脈調整比值 (即 0.679) 不小於最小時脈調整比值 (即 0.6)，所以最小時脈調整比值仍保持為 0.6，接著會將鬆弛參數修正為 14 (即  $9+5$ )，以加入工作  $\tau_2$  的工作執行時間。而此時的閒置結束參數 (即 28) 不小於工作  $\tau_2$  的完成工作時限  $D_2$  (即 28)，所以會往下執行步驟 S404。至此，可計算出工作  $\tau_2$  的最小時脈調整比值 (即 0.6)。

除此之外，可再提一實施例，以使本發明的實施例更加容易了解。圖 6A 為根據本發明再一實施例的頻率調整方法的流程圖。請參照圖 3B 及圖 6A，本實施例是以演算法的流程型態來呈現，且為了解說方便，在此仍以上述三個工作  $\tau_1 \sim \tau_3$  來說明，所以演算法中的  $n=3$ 。在本實施例中，首先步驟 S610 會將陣列型態的閒置結束參數 `idleEnd` 及鬆弛參數 `slack` 內的數值皆設為 0。接著，在步驟 S620 中，會由工作  $\tau_3$  到  $\tau_1$  依序各別呼叫函式 `Energy-Minimizing-Freq02` 以計算工作  $\tau_1 \sim \tau_3$  的最小時脈調整比值  $\varepsilon_1 \sim \varepsilon_3$ ，亦即由最低優先權順序的工作  $\tau_3$  開始計算其最小時脈調整比值  $\varepsilon_3$ 。

最後，步驟 S630 會利用最小時脈調整比值  $\varepsilon_1 \sim \varepsilon_3$  計算出工作  $\tau_1 \sim \tau_3$  的理想頻率  $\nu_1 \sim \nu_3$ ，以供處理器 210 調整工作  $\tau_1 \sim \tau_3$  對應的工作頻率。而步驟 S631 會判斷工作  $\tau_2 \sim \tau_3$  的

其中之一的理想頻率是否小於其前一工作的理想頻率，以在工作  $\tau_2 \sim \tau_3$  的其中之一的理想頻率小於其前一工作的理想頻率時，依據優先權順序高於此工作的這些工作的理想頻率，重新計算此工作的最小時脈調整比值  $\varepsilon_j$ ，以藉此調整此工作的理想頻率。其中，函式 Inflated-Energy-Minimizing-Freq 可以為先前技術中論文 "Practical voltage-scaling for fixed-priority RT-systems" 所述相關部份來實施，在此則不再加以贅述，以及函數  $lp$  為優先權順序低於等於工作  $\tau_i$  的工作的集合。

接著，再對函式 Energy-Minimizing-Freq02 作進一步說明，其中部份函數可參照函式 Energy-Minimizing-Freq01 的說明。圖 6B 為圖 6A 的函式 Energy-Minimizing-Freq02 的流程圖。請參照圖 3B 及圖 6B，在步驟 S621 中，會設定所有參數值的初始值，參數  $S$ 、 $I$ 、 $\beta$ 、 $\Delta$  及  $\omega$  設定為零，參數  $\alpha$ （最小時脈調整比值）及  $idleIdx$  設為 1，參數  $IN\_BZP$  設定為「TRUE（真）」，參數  $\omega$  則設定為工作執行時間  $C_i$  除以最大使用頻率  $f_{max}$ （此時為計算工作  $\tau_3$ ，亦即此次的參數  $\omega$  為  $C_3/f_{max}=3$ ）。接著，在步驟 S622 中，會判斷所計算的工作是否為最低優先權的工作  $\tau_3$ 。

由於本實施是從最低優先權的工作  $\tau_3$  開始計算，所以步驟 S622 第一次執行的結果會為「Yes」，接著會執行步驟 S623，以判斷參數  $IN\_BZP$  是否為「TRUE」。由於參數  $IN\_BZP$  的初始值為「TRUE」，且在此之前並未更動參數  $IN\_BZP$  的值，所以步驟 S623 第一次執行的結果會為「Yes」，接著執行步驟 S624 以計算工作  $\tau_3$  閒置時間的起始點。在步驟 S624 中，步驟 S625 第一次執行會得到參數  $\omega=15$ （此為第一個閒置時間的起始點），參數  $\Delta=12$ ，由於參數  $\Delta$  不為零，所以要再執行一

次以驗證參數 $\omega$ 是否正確。其中，函數 hp 為優先權順序高於等於工作 $\tau_i$ 的工作的集合，函數 floor 為取得小於括號中算式計算結果的最大整數。而步驟 S625 第二次執行的結果仍得到參數 $\omega=15$ ，而參數 $\Delta=0$ ，代表計算出的參數 $\omega$ 正確，接著將參數 IN\_BZP 設定為「FALSE (偽)」，以進行時脈調整比值的計算。接著在步驟 S628 中，因為參數 $\omega$  (即 15) 小於工作 $\tau_3$ 的完成工作時限 $D_3$  (即 30)，所以會回到步驟 S622。

由於此時函式 Energy-Minimizing-Freq02 係被呼叫以計算工作 $\tau_3$ ，且此時的參數 IN\_BZP 已被更改為「FALSE」，所以會經由步驟 S622、S623，再進行到步驟 S626。在步驟 S626 中，會得到 $\beta/\omega$  (亦即時脈調整比值) $=15/20$  (即 0.75)，由於 $\beta/\omega$  (即 0.75) 小於參數 $\alpha$ 的初始值 (即 1)，所以參數 $\alpha$ 會更新為 0.75，其中函數 ceil 為取得大於括號中算式計算結果的最小整數。同時，閒置結束參數 idleEnd<sub>1</sub> 會設定為 20，鬆弛參數 slack<sub>1</sub> 會設定為 8 (即 5+3)，並將 IN\_BZP 設定為「TRUE」，以計算下一個時脈調整比值，以及將 idleIdx 增加為 2。以下為了說明書的簡潔，則省略判斷式的詳細說明。

此時的函式 Energy-Minimizing-Freq02 會回到步驟 S622，由於 IN\_BZP 又被設定為「TRUE」，所以流程會回到進行步驟 S624。在執行步驟 S624 後會得到參數 $\omega=27$  (此為第二個閒置時間的起始點)，並將 IN\_BZP 設定為「FALSE」。接著會執行步驟 S626，會得到 $\beta/\omega$  (亦即時脈調整比值) $=22/28$  (即 0.786)，閒置結束參數 idleEnd<sub>2</sub> 會設定為 28，鬆弛參數 slack<sub>2</sub> 會設定為 9 (即 6+3)。因為此時的 $\beta/\omega$  (即 0.786) 大於參數 $\alpha$  (即 0.75)，所以參數 $\alpha$ 不會被更動，再將 IN\_BZP 更改為「TRUE」。而函式 Energy-Minimizing-Freq02 此時會經

由步驟 S628、S622 及 S623，再回到步驟 S624。而此次步驟 S624 的執行結果會得到參數  $\omega=33$ （不小於工作  $\tau_3$  的完成工作時限 30），代表接下來的閒置時間超出工作  $\tau_3$  的完成工作時限 30，致使函式 Energy-Minimizing-Freq02 的流程在執行到步驟 S628 時，會進行回傳參數  $\alpha$  到演算法主體的步驟。

接著函式 Energy-Minimizing-Freq02 會被再次呼叫以計算工作  $\tau_2$  的最小時脈調整比值，步驟 S621 的執行如上述說明，在此則不在贅述。而執行步驟 S622 時，會由於工作  $\tau_2$  非最低優先權順序的工作，所以會接著執行步驟 S627。在步驟 S627 中，此時會取得閒置結束參數  $idleEnd_1$ （即 20）及鬆弛參數  $slack_1$ （即 8），分別存入參數  $\omega$  及  $S$ ，而得到  $\beta/\omega=12/20$ （即 0.6）。由於  $\beta/\omega$ （即 0.6）小於參數  $\alpha$  的初始值（即 1），所以參數  $\alpha$  會更新為 0.6，並且鬆弛參數  $slack_1$  會修正為 13（即  $8+5$ ），以及  $idleIdx$  會增加為 2。由於此時的參數  $\omega$ （即 20）小於工作  $\tau_2$  的完成工作時限  $D_2$ （即 28），致使函式 Energy-Minimizing-Freq02 會再回到執行步驟 S627。

而再次執行步驟 S627 時，會取得閒置結束參數  $idleEnd_2$ （即 28）及鬆弛參數  $slack_2$ （即 9），同樣分別存入參數  $\omega$  及  $S$ ，而得到  $\beta/\omega=19/28$ （即 0.679）。因為此時的  $\beta/\omega$ （即 0.679）大於參數  $\alpha$ （即 0.6），所以參數  $\alpha$  不會被更動，並且參數  $\omega$ （即 28）不小於工作  $\tau_2$  的完成工作時限  $D_2$ （即 28），致使函式 Energy-Minimizing-Freq02 會結束計算並且回傳參數  $\alpha$  到演算法主體中。

再接著，函式 Energy-Minimizing-Freq02 會再被呼叫以計算工作  $\tau_1$  的最小時脈調整比值，在步驟 S627 中，會取得閒置結束參數  $idleEnd_1$ （即 20）及鬆弛參數  $slack_1$ （即 13），分別

存入參數  $\omega$  及  $S$ ，而得到  $\beta/\omega=7/20$  (即 0.35)。由於  $\beta/\omega$  (即 0.35) 小於參數  $\alpha$  的初始值 (即 1)，所以參數  $\alpha$  會更新為 0.35，並且鬆弛參數  $\text{slack}_1$  會更新為 20 (即 13+7)，以及  $\text{idleIdx}$  會增加為 2。由於此時的參數  $\omega$  (即 20) 不小於工作  $\tau_1$  的完成工作時限  $D_1$  (即 20)，所以函式 Energy-Minimizing-Freq02 會結束計算並且回傳參數  $\alpha$  到演算法主體中。至此，可計算出工作  $\tau_1 \sim \tau_3$  的最小時脈調整比值分別為 0.35、0.6、0.75，使得執行圖 6A 的步驟 S630 時，可以依據這些最小時脈調整比值計算出工作  $\tau_1 \sim \tau_3$  的理想頻率，以調整工作  $\tau_1 \sim \tau_3$  各自的工作時脈。

在此，可以針對傳統函式 Energy-Minimizing-Freq01 及本實施例的 Energy-Minimizing-Freq02 作比對。請參照圖 1 及圖 6B，由圖中可以得到，步驟 S130 會等於步驟 S624 (亦即步驟 S131 會等於步驟 S625)，步驟 S140 與步驟 S626 的計算量可視為相同 (因其間的差異與「 $I \leftarrow \min_{j \in \text{hp}(\tau_i)} \left\{ \left( T_j^* \left\lceil \frac{\omega}{T_j} \right\rceil \right) - \omega, D_i - \omega \right\}$ 」相較甚微，故可忽略不計)，而步驟 S627 的計算式與步驟 140 相似，但計算量少於步驟 140 (因步驟 140 中的「 $I \leftarrow \min_{j \in \text{hp}(\tau_i)} \left\{ \left( T_j^* \left\lceil \frac{\omega}{T_j} \right\rceil \right) - \omega, D_i - \omega \right\}$ 」相當耗時，會主宰計算時間較鉅)。以傳統函式 Energy-Minimizing-Freq01 而言，計算工作  $\tau_1$  的最小時脈調整比值會執行 1 次步驟 S131，1 次步驟 S140；計算工作  $\tau_2$  的最小時脈調整比值會執行 4 次步驟 S131，2 次步驟 S140；計算工作  $\tau_3$  的最小時脈調整比值會執行 5 次步驟 S131，2 次步驟 S140。綜上述，步驟 S131 被執行 10 次，步驟 S140 被執行 5 次。

以本實施例的函式 Energy-Minimizing-Freq02 而言，計算工作  $\tau_3$  的最小時脈調整比值會執行 5 次步驟 S625，2 次步驟

S626；計算工作  $\tau_2$  的最小時脈調整比值會執行 2 次步驟 S627；計算工作  $\tau_1$  的最小時脈調整比值會執行 1 次步驟 S627。總而言之，步驟 S625 被執行 5 次，步驟 S626 被執行 2 次，步驟 S627 被執行 3 次。比較上述函式的運算量，並彙整如表 1 所示，其中步驟 S131 及 S140 的運算量分別以 [a1] 及 [a2] 來表示，而步驟 S625、S626 及 S627 的運算量分別以 [b1]、[b2]、[b3] 來表示。由表 1 可得知，本實施例的函式 Energy-Minimizing-Freq02 的計算量會低於傳統函式 Energy-Minimizing-Freq01。因此，本實施例的函式 Energy-Minimizing-Freq02 可以更快計算出最佳排程，以加快計算速度，減少計算的時間。

表 1

函式 運算量 工作	Energy-Minimizing-Freq01	Energy-Minimizing-Freq02
工作 $\tau_1$	[a1], [a2]	[b3]
工作 $\tau_2$	4*[a1], 2*[a2]	2*[b3]
工作 $\tau_3$	5*[a1], 2*[a2]	5*[b1], 2*[b2]
總和	10*[a1], 5*[a2]	5*[b1], 2*[b2], 3*[b3]

此外，若硬體可以支援調整即時系統的工作電壓時，可以在調變即時系統之處理器的工作時脈的同時，據以調整即時系統的工作電壓，並且工作時脈與工作電壓的調變方式為：(1) 調降工作時脈與工作電壓時，先調降工作時脈後，再調降工作電壓。(2) 調升工作時脈與工作電壓時，先調升工作電壓後，再調升工作時脈。



綜上所述，本發明實施例的頻率調整方法及電子裝置，透過計算最低優先權工作的過程中，取得運算資訊中的閒置結束參數及對應鬆弛參數，再利用閒置結束參數及鬆弛參數計算出非最低優先權工作的最小時脈調整比值。藉此，可節省耗時的計算過程，以加速最佳排程的計算速度，減少最佳排程的計算時間。

本領域具通常知識者可以依其需求，而以任何形式實現上述諸實施例。例如，可以內儲程式之電腦可讀取儲存媒體來實現上述諸實施例。也就是說，上述諸實施例可以用程式(電腦軟體)形式實現之，而將此程式儲存在硬碟、軟碟、CD-ROM等電腦可讀取儲存媒體中。

由於網路之普及，上述程式(電腦軟體)除可儲存於電腦可讀取儲存媒體外，亦可在網路上直接傳輸提供，而無須藉由儲存於儲存媒體上提供該程式。因此，本領域具通常知識者可以依其需求，而以「電腦程式產品」型式實現上述諸實施例。所謂電腦程式產品，係載有電腦可讀取之程式且不限外在形式之物。

雖然本發明已以實施例揭露如上，然其並非用以限定本發明，任何所屬技術領域中具有通常知識者，在不脫離本發明之精神和範圍內，當可作些許之更動與潤飾，故本發明之保護範圍當視後附之申請專利範圍所界定者為準。

### 【圖式簡單說明】

圖 1 為傳統實施例的函式 Energy-Minimizing-Freq01 的流程圖。

圖 2 為根據發明一實施例之電子裝置的系統方塊圖。

圖 3A 為根據本發明一實施例之頻率調整方法的流程圖。

圖 3B 為圖 3A 之工作排程示意圖。

圖 4 為根據本發明另一實施例的頻率調整方法的流程圖。

圖 5 為圖 4 步驟 S403 中精簡演算法的細部流程圖。

圖 6A 為根據本發明再一實施例的頻率調整方法的流程圖。

圖 6B 為圖 6A 的函式 Energy-Minimizing-Freq02 的流程圖。

### 【主要元件符號說明】

S110、S120、S130、S131、S140、S150：傳統實施例的函式 Energy-Minimizing-Freq01 的步驟

200：電子裝置

210：處理器

220：記憶單元

S301~S304、S401~S407、S610、S620、S630、S631：本發明實施例的頻率調整方法的步驟

S501~S507：精簡演算法的步驟

S621~S628：本發明實施例的函式 Energy-Minimizing-Freq02 的步驟

$\tau_1 \sim \tau_3$ ：工作

## 七、申請專利範圍：

### 1.一種頻率調整方法，包括：

取得在一即時系統之一工作排程中多個工作所對應的工作資訊，其中該些工作各自使用對應的工作時脈；

依據該工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算該些工作各自的最小時脈調整比值，其中使用一完整演算法去計算該些工作中最低優先權者的最小時脈調整比值並記錄其中運算資訊，以及使用一精簡演算法與前述運算資訊去計算該些工作中非最低優先權者的最小時脈調整比值；

依據該些工作的最小時脈調整比值以及該些工作的優先權順序，計算出該些工作各自的參考頻率；以及

依據該些工作的參考頻率，調整該些工作所對應的工作時脈。

2.如申請專利範圍第1項所述之頻率調整方法，其中工作所對應的工作資訊包括工作執行時間、工作觸發週期以及完成工作時限。

3.如申請專利範圍第1項所述之頻率調整方法，其中使用對應的工作資訊計算該些工作各自的最小時脈調整比值的步驟包括：

利用該完整演算法計算該些工作中最低優先權順序的第一工作的最小時脈調整比值，並且記錄該運算資訊中至少一閒置結束 (idle end) 參數及其對應的多個鬆弛 (slack) 參數，其中所述閒置結束參數為該第一工作的一完成工作時限內至少一閒置時間的結束時間點，而所述鬆弛參數為所述閒置時間的時間長度；以及

依據所述閒置結束參數及其對應的鬆弛參數，利用該精簡演算法計算該些工作中一第二工作的最小時脈調整比值，其中該第二工作的優先權順序高於該第一工作。

4.如申請專利範圍第3項所述之頻率調整方法，其中該精簡演算法包括：

設定該最小時脈調整比值的初始值為1；

依序取得該些閒置結束參數的其中之一及其對應的鬆弛參數；

利用該閒置結束參數及其對應的鬆弛參數依序計算出該第二工作的一時脈調整比值；

若該時脈調整比值小於該最小時脈調整比值時，則記錄該時脈調整比值作為該第二工作的最小時脈調整比值；

依據該第二工作的一工作執行時間調整該鬆弛參數；以及在該閒置結束參數小於該第二工作的該完成工作時限時，再次進行該精簡演算法。

5.如申請專利範圍第1項所述之頻率調整方法，其中計算出該些工作的參考頻率的步驟包括：

依據該些工作由優先權最高至優先權最低的順序，依序從優先權順序最高的該工作進行計算，且每一該些工作的參考頻率是取優先權順序低於或等於其優先權順序之該些工作的最小時脈比值的最大值。

6.如申請專利範圍第1項所述之頻率調整方法，更包括：

在該些工作的其中之一的參考頻率小於其前一工作的參考頻率時，依據優先權順序高於該工作的該些工作的參考頻率，調整該工作的參考頻率。

7.如申請專利範圍第 1 項所述之頻率調整方法，其中調整該些工作其中之一所對應的工作時脈的步驟包括：

當調降該即時系統之處理器的一工作電壓與該即時系統之處理器的該工作時脈時，先調降該工作時脈後，再調降該工作電壓；以及

當調升該即時系統之處理器的該工作電壓與該即時系統之處理器的該工作時脈時，先調升該工作電壓後，再調升該工作時脈。

8.一種電腦可讀取儲存媒體，用以儲存一電腦程式，該電腦程式用以載入至一電腦系統中並且使得該電腦系統執行如申請專利範圍第 1 項所述頻率調整方法。

9.一種電腦程式產品，當電子裝置載入該電腦程式並執行後，可完成申請專利範圍第 1 項所述之頻率調整方法。

10.一種電子裝置，包括：

一處理器，依據該些工作對應的工作時脈而運行，其中該處理器取得在一即時系統之一工作排程中多個工作所對應的工作資訊，且依據該工作排程之優先權順序，以優先權最低者至優先權最高者的順序，依序使用對應的工作資訊計算該些工作各自的最小時脈調整比值，然後依據該些工作的最小時脈調整比值以及該些工作的優先權順序，計算出該些工作各自的參考頻率，接著依據該些工作的參考頻率，調整該些工作所對應的工作時脈；其中，該處理器使用一完整演算法去計算該些工作中最低優先權者的最小時脈調整比值並記錄其中運算資訊，以及使用一精簡演算法與前述運算資訊去計算該些工作中非最低優先權者的最小時脈調整比值；以及

一記憶單元，耦接於該處理器，儲存該些工作的工作資訊、該些最小時脈調整比值，以及該些參考頻率。

11.如申請專利範圍第 10 項所述之電子裝置，其中該處理器利用該完整演算法計算該些工作中最低優先權順序的一第一工作的最小時脈調整比值，並且記錄運算資訊中至少一閒置結束 (idle end) 參數及其對應的鬆弛 (slack) 參數，以及依據所述閒置結束參數及對應的鬆弛參數，利用該精簡演算法計算該些工作中一第二工作的最小時脈調整比值；其中，所述閒置結束參數為一閒置時間的結束時間，所述鬆弛參數為該閒置時間的長度，且該第二工作的優先權順序高於該第一工作。

12.如申請專利範圍第 10 項所述之電子裝置，其中該處理器更執行該精簡演算法，以設定該最小時脈調整比值的初始值為 1，接著依序取得所述閒置結束參數的其中之一及其對應的鬆弛參數以利用該閒置結束參數及其對應的鬆弛參數依序計算出該第二工作的一時脈調整比值，在該時脈調整比值小於該最小時脈調整比值時，記錄該時脈調整比值作為該第二工作的最小時脈調整比值，以及依據該第二工作的一工作執行時間調整該鬆弛參數，並且在該閒置結束參數小於該第二工作的一完成工作時限時，更次執行該精簡演算法。

八、圖式：

29129TW\_M

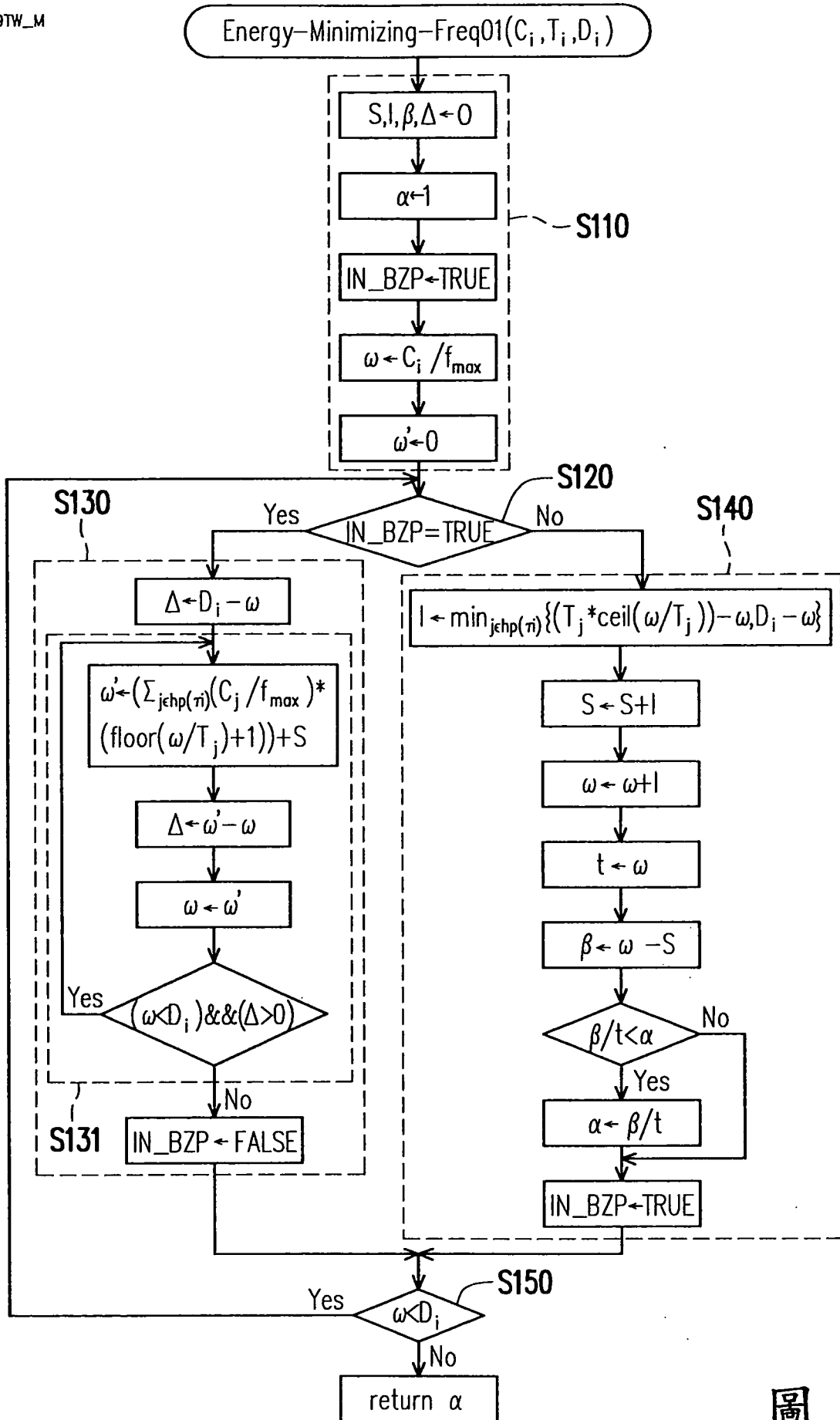


圖 1

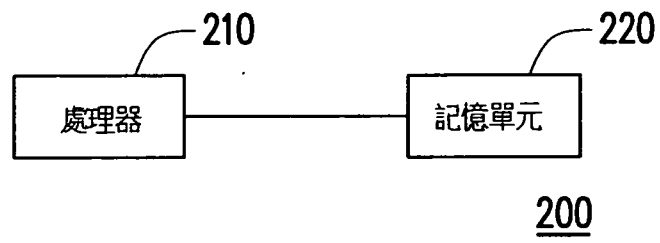


圖 2



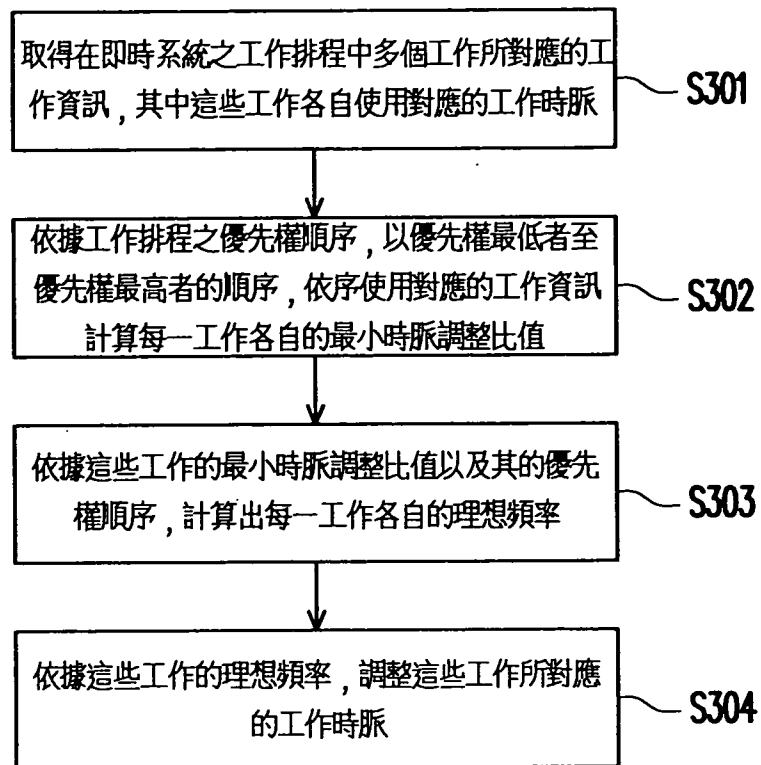


圖 3A

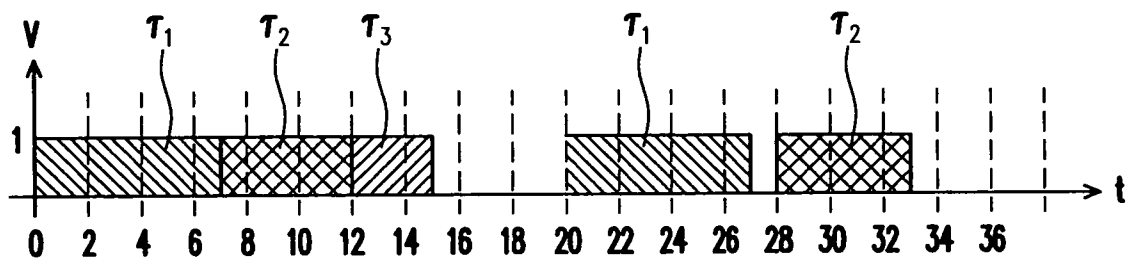


圖 3B

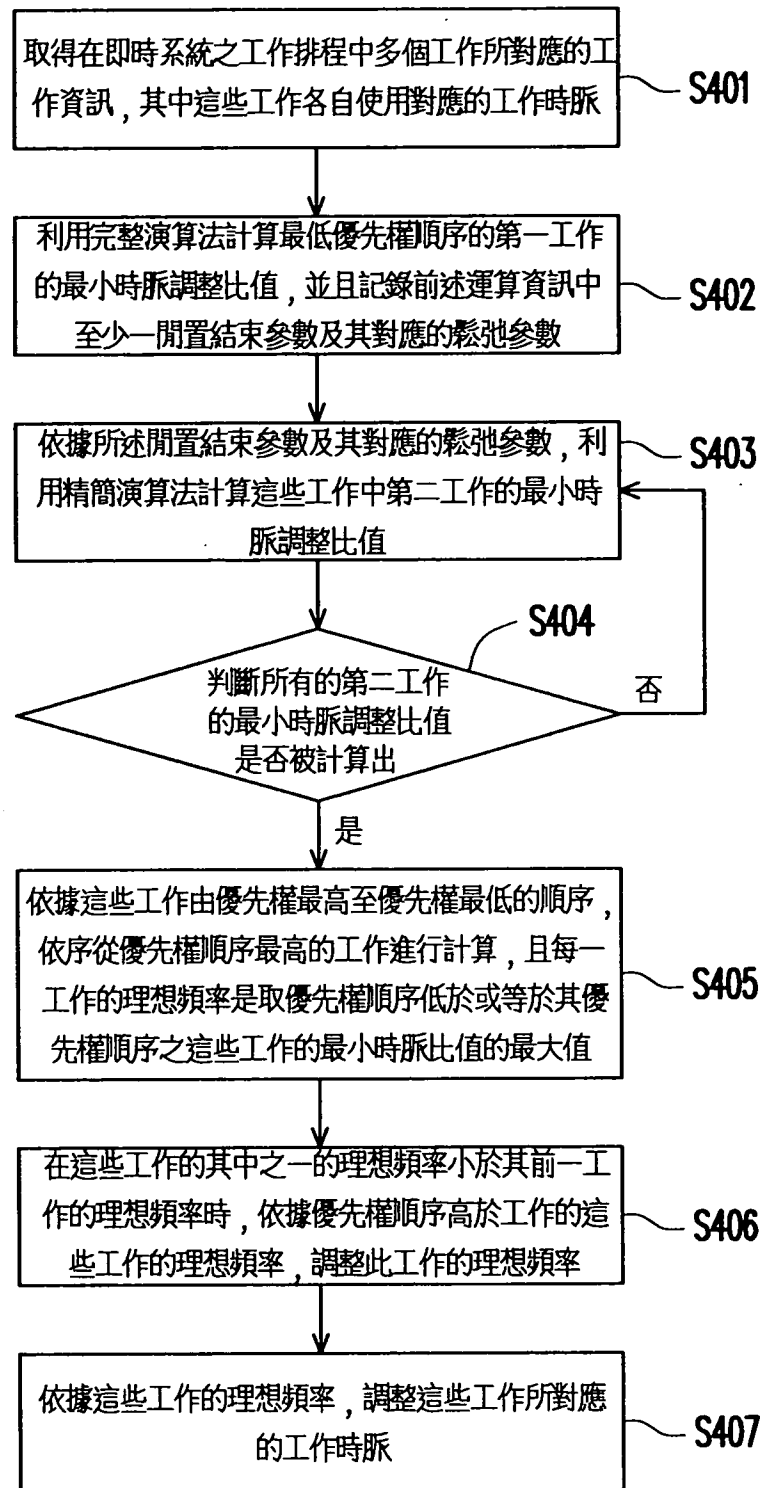


圖 4

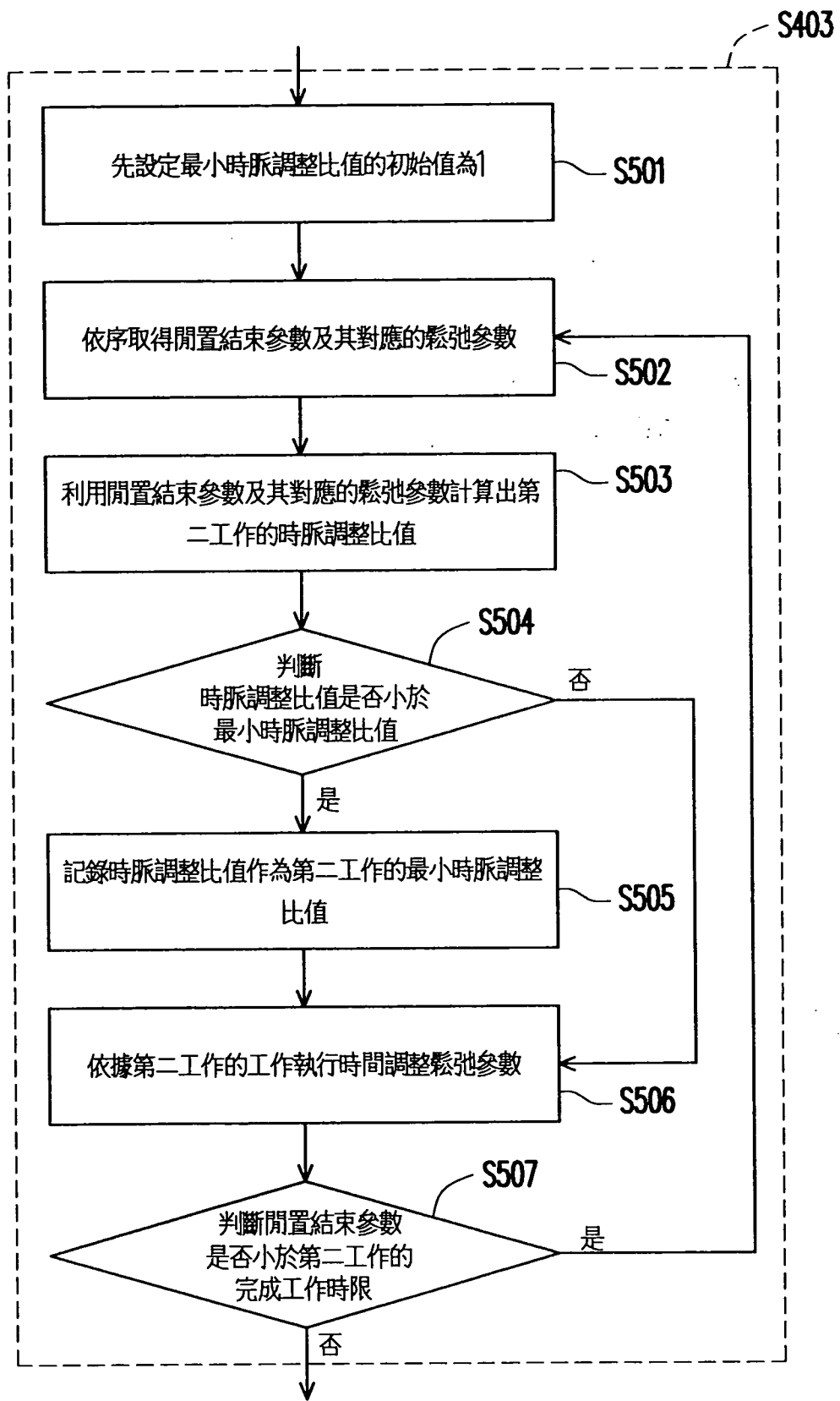


圖 5

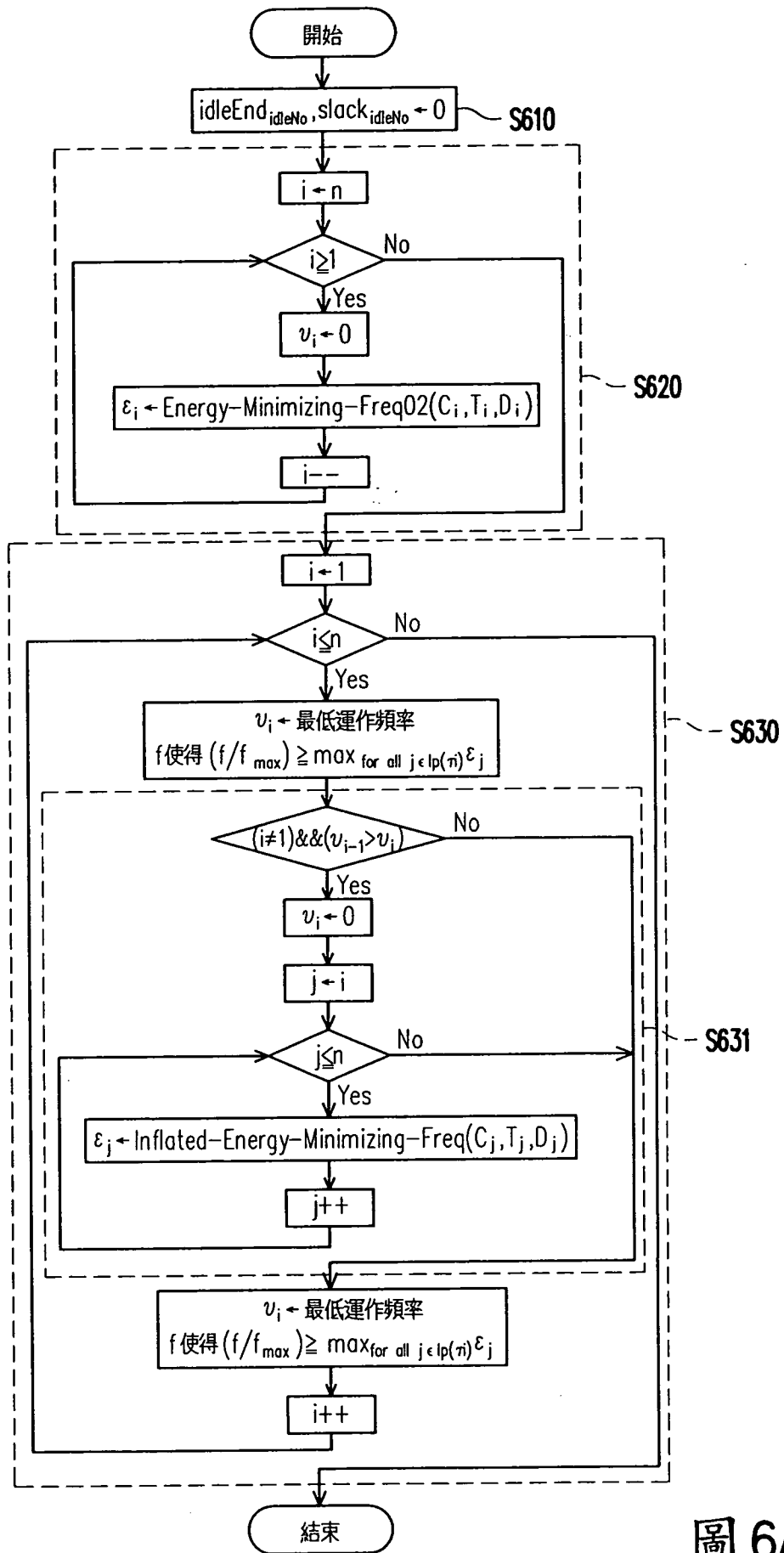


圖 6A

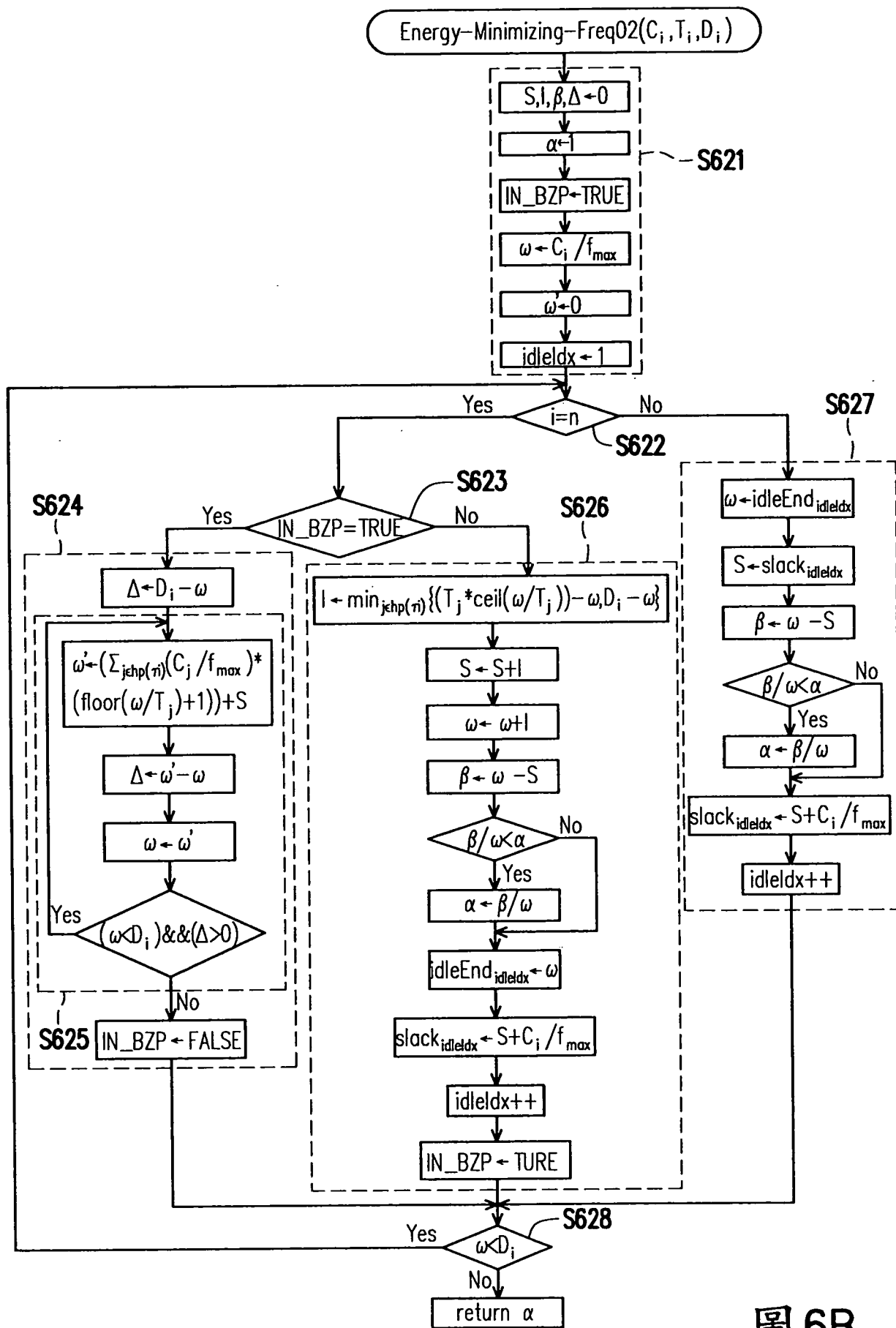


圖 6B