

## 公告本

737987

|      |               |
|------|---------------|
| 申請日期 | 89 年 9 月 30 日 |
| 案 號  | 89120417      |
| 類 別  | G06F 9/28     |

A4  
C4

499656

(以上各欄由本局填註)

## 發明專利說明書

|             |               |                                                                                                                                                                                                                                              |
|-------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 一、發明<br>名稱  | 中 文           | 浮點指令組結構及其應用方法                                                                                                                                                                                                                                |
|             | 英 文           | A floating point instruction set architecture and implementation                                                                                                                                                                             |
| 二、發明<br>創作人 | 姓 名           | (1) 席瓦倫·克里斯能 Krishnan, Sivaram<br>(2) 馬克·迪其格 Debbage, Mark<br>(3) 肯納德·羅伊 Roy, Kanad                                                                                                                                                          |
|             | 國 籍           | (1) 美國 (2) 英國 (3) 加拿大                                                                                                                                                                                                                        |
|             | 住、居所          | (1) 美國加州洛沙特斯西布魯克大道一七二三號<br>1723 Westbrook Avenue Los Altos, California 94024-5321 USA<br>(2) 美國加州聖喬西南一一街三九八號<br>398 S. 11th St. San Jose, California 95112 USA<br>(3) 美國加州聖喬斯盧貴諾圓環二九一五號<br>2915 Rubino Circle San Jose, California 95125 USA |
| 三、申請人       | 姓 名<br>(名稱)   | (1) 日立製作所股份有限公司<br>株式会社日立製作所                                                                                                                                                                                                                 |
|             | 國 籍           | (1) 日本                                                                                                                                                                                                                                       |
|             | 住、居所<br>(事務所) | (1) 日本國東京都千代田區神田駿河台四丁目六番地                                                                                                                                                                                                                    |
|             | 代 表 人<br>姓 名  | (1) 庄山悅彦                                                                                                                                                                                                                                     |

裝

訂

線

|      |               |
|------|---------------|
| 申請日期 | 89 年 9 月 30 日 |
| 案 號  | 89120417      |
| 類 別  |               |

A4  
C4

( 以上各欄由本局填註 )

| 發 明 專 利 說 明 書 |               |                                                                                                                                                                                                                                                         |
|---------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 一、發明<br>名稱    | 中 文           |                                                                                                                                                                                                                                                         |
|               | 英 文           |                                                                                                                                                                                                                                                         |
| 二、發明<br>人     | 姓 名           | (4) 荒川文男 Arakawa, Fumio<br>(5) 安帝·史特格 Sturges, Andy Craig<br>(6) 葛蘭·菲瑞爾 Farrall, Glenn Ashley                                                                                                                                                           |
|               | 國 籍           | (4) 日本 (5) 英國 (6) 澳洲                                                                                                                                                                                                                                    |
|               | 住、居所          | (4) 日本國東京都小平市津田町一一一四一一九<br>1-14-19 Tsudamachi, Kodaira-shi, Tokyo 187-0025, Japan<br>(5) 英國貝斯貝佛德幫浦巷三號<br>3 Pump Lane, Bathford, Bath, United Kingdom<br>(6) 英國布魯斯特長皮斯頓兒童巷小木屋<br>The Cottage, Keeds Lane Long Ashton, Bristol BS41 9BY<br>United Kingdom |
|               | 三、申請人         | 姓 名<br>(名稱)                                                                                                                                                                                                                                             |
|               | 國 籍           |                                                                                                                                                                                                                                                         |
|               | 住、居所<br>(事務所) |                                                                                                                                                                                                                                                         |
|               | 代 表 人<br>姓 名  |                                                                                                                                                                                                                                                         |

裝  
訂  
線

經濟部智慧財產局員工消費合作社印製

(由本局填寫)

|           |
|-----------|
| 承辦人代碼：    |
| 大類：       |
| I P C 分類： |

A6  
B6

本案已向：

國（地區）

申請專利，申請日期：

案號：

，☐有 ☐無主張優先權

美國

1999 年 10 月 1 日

09/411,600

☒有主張優先權

有關微生物已寄存於：

，寄存日期：

，寄存號碼：

（請先閱讀背面之注意事項再填寫本頁各欄）

裝  
訂  
線

經濟部智慧財產局員工消費合作社印製

五、發明說明 ( 1 )

發明背景

本發明一般有關微處理器／微控制器結構，特別有關一浮點組結構。

圖 1 說明一電腦系統的不同層級及顯示使用軟體和硬體以供微處理器或中央處理單元 ( C P U ) 設計及實施之優缺點取捨。如圖 1 所示，將微處理器功能傾向硬體一般會增加速度 6，但減低彈性 8，而將硬體功能以軟體取代一般增加使用彈性 8，但代價為降低速度 6。

在軟體中之微程式之引入使更多的硬體功能可在軟體中動作。結果，指令組在組合語言層 1 6 變得更複雜。由於在硬體成本的減少，藉由某些軟體功能被執行於特殊硬體，例如浮點處理器，使平衡點移回硬體。如此容許效能之提升。

直到最近這些浮點副處理器，以 1 6 位元或 3 2 位元執行運算，包括浮點算術及比較。藉由 6 4 位元微處理器的發展，6 4 位元副處理器亦被發展成型。但是，指令組及記憶體一般是以 3 2 位元資料字元為基礎。對於使用此 3 2 位元資料字元的 6 4 位元副處理器上的浮點算術，需要一在單精準度 ( 3 2 位元 ) 資料字元的記憶體之間的一載入／儲存，及在雙精準度 ( 6 4 位元 ) 資料字元的記憶體之間的兩個載入／儲存作業。如此，假設一字元邊界為記憶體每 3 2 位元，新 6 4 位元 C P U 及副處理器被不足應用於資料傳送。

同時對浮點比較而言，因為浮點單元被考慮為與

## 五、發明說明 ( 2 )

C P U 分離之單元，比較結果被寫入至一浮點暫存器或一條件碼被設定。此條件碼接著被用於 C P U 中的條件指令，需要 C P U 中之額外邏輯。就整數比較而言，比較結果有時被寫入至一整數暫存器，需要一整數指令以測試與該浮點比較不同的暫存器。此種系統在不同地方必需檢查不同暫存器；根據一浮點或整數指令是否被執行，是不利的。

因此，需要一種浮點指令組，其利用 6 4 位元結構對 3 2 位元或 6 4 位元資料值更為有效率，特別是，例如，在載入及儲存作業中，以及其在浮點比較運算中與 C P U 有更有效率之界面。

### 發明概要

大體而言，本發明係有關處理器及副處理器，各被構建以執行一浮點指令組結構。更特別地，舉例說明一浮點副處理器及記憶體之間的雙精準度字元及／或單精準度字元對的載入及儲存。亦舉例說明在一 C P U 中之一整數暫存器中之一浮點單元中的兩個浮點數之布林比較結果的儲存。

在一特定實施例中，本發明提供一方法用以將資料自一記憶體載入至一中央處理單元 ( C P U )，包括由一電腦指令中的一運算碼決定一載入作業及由一 K 位元寬 N 群組之資料載入至 C P U 中之每一群組具有 M 位元的一儲存位置。K，M 及 N 為整數。在 C P U 中之儲存位置可為一

五、發明說明 ( 3 )

6 4 位元暫存器。如所舉例，K 等於 3 2，N 等於 1，及 M 等於 8 位元，或 K 等於 3 2，N 等於 2，M 等於 4 位元。

在一特定實施例中，本發明亦提供一方法用以自一記憶體儲存資料至一 CPU，包括步驟如由一電腦指令中的一運算碼決定是否有一儲存作業，且依照運算碼從 CPU 中之儲存位置將 N 群組資料儲存至 K 位元寬記憶體，其中每一群組包含 M 位元。

在另一實施例中，本發明提供一方法用以在一電腦系統中比較多數浮點數，包括步驟如決定在一電腦指令中是否一運算碼解碼以供一比較運算。接著藉由將多數浮點數的第一浮點數與多數浮點數的第二浮點數作比較，且將布林結果儲存在一儲存位置。儲存位置較宜為一般目的暫存器 ( G P R )，其可包括一整數暫存器。再者，該結果可被用於一條條件分支指令，例如，如果 - 則 - 或者 ( if-then-else )。在作業上，布林結果可由一群包含位元一及零中選出。

本發明的這些及其它優點及特色將可為熟習該技藝人士於讀過以下詳細敘述後清楚了解，其中將參照所附圖示另以解說。

圖式概要說明

圖 1 顯示習知技術電腦系統的不同層級；

圖 2 顯示本發明的記憶體及浮點暫存器之間的 6 4 位

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

五、發明說明 ( 4 )

元資料交換的特定實施例；

圖 3 顯示本發明的兩個浮點暫存器的內容的邏輯比較的布林結果的特定實施例；

圖 4 顯示 S 5 核心的頂層分隔的一例子；

圖 5 顯示在 C P U 中之有限數區塊的一特定實施例。

主要元件對照表

|               |           |
|---------------|-----------|
| 1 1 0         | 記憶體       |
| 1 2 0         | D R i 暫存器 |
| 1 2 6 , 1 2 8 | 暫存器       |
| 1 4 2 , 1 4 4 | F P 暫存器   |
| 2 4 0         | 載入儲存單元    |
| 2 0 5         | 匯流排界面單元   |
| 2 1 0         | 指令流單元     |
| 2 2 0         | 指令多媒體單元   |
| 2 3 0         | 指令快取單元    |
| 2 5 0         | 資料快取單元    |
| 2 6 5         | 浮點單元      |

較佳實施例的敘述

浮點指令實例

在本發明的一特定實施例中，附件 1 中為一浮點指令實例之表列，附件 2 為浮點記憶體指令的一實例。從指令表列中，數個表列被敘述以說明浮點指令組的特色。

## 五、發明說明 ( 5 )

在一特定實施例中在電腦系統中的一記憶體為定義以 32 位元 ( 4 位元組 ) 字元邊界的位元組。載入及儲存指令可使用小結尾或大結尾代表而提供對資料之存取。載入及儲存指令之結尾係在開機重設中指明，其後不再改變。

對於此段中的例子而言，每一指令包含 32 位元。這些位元被分組為連續接收位元，稱為位元欄。每一位元欄與一位元欄類型相關。例如有一運算碼之欄類形，用以辨別暫存器，以編碼一中介元及用於移位。在一特定實施例中，在一指令中的位元欄使用小結尾二進位代表被編碼。在位元欄中的最高 ( 最左邊 ) 位元為最重要，在位元欄中之最低位元數為最不重要。一位元欄係依位元欄類型所指明而被解讀為未標示或標示。數字係成二之補數形式。

圖 2 顯示本發明之在記憶體及浮點暫存器之間的 64 位元資料的交換的一特定實施例。記憶體 110，例如，包括位址 0 - 7 的 8 位元：D0 至 D7，以小結尾格式表現。記憶體 110 為 32 位元寬 112，亦即，字元邊界為每 4 位元組。

若記憶體 110 包在位址 0 - 7 中包含一雙精準度字元 ( 64 - 位元 )，則一特定實施例包括一指令其在浮點單元 ( FPU ) 中自記憶體 110 係雙精準度字元載入至一雙精準度暫存器，亦即，DRi 暫存器 ( 其中 i 為一整數 ) 120。資料 D0 - D7，來自記憶體 110，被顯示於 DRi 暫存器 120 中。在 DRi 120 中的雙精準度字元亦可被儲存在記憶體 110 的位址 0 - 7。在另一

五、發明說明 ( 6 )

實施例中在記憶體 1 1 0 中的雙精準度字元可被載入至兩個單精準度暫存器，亦即暫存器  $F P 2 i + 1 1 2 6$  及暫存器  $F P 2 i 1 2 8$ 。當中每一單精準度暫存器為 3 2 位元，6 4 位元資料的最重要一半；亦即， $D 4 - D 7$ ，被儲存在  $F P 2 i + 1 1 2 6$ ；且 6 4 位元資料的最不重要一半，亦即  $D 0 - D 3$ ，被儲存在  $F P 2 i 1 2 8$ 。同時反向程序可接著進行以將  $F P 2 i + 1 1 2 6$  及  $F P 2 i 1 2 8$  中的雙精準度字元儲存至記憶體 1 1 0 之位址 0 - 7。

若記憶體 1 1 0 包含在位址 0 - 3，亦即  $D 0 - D 3$ ，及位址 4 - 7，亦即  $D 4 - D 7$  的兩個單精準度字元，則本發明的一特定實施例包括一指令，其在浮點單元 (F P U) 中將單精準度字元對向記憶體載入至兩個單精準度暫存器，亦即暫存器  $F P 2 i + 1 1 2 6$  及暫存器  $F P 2 i 1 2 8$ 。如此 6 4 位元資料的最重要一半，亦即  $D 4 - D 7$  被儲存在  $F P 2 i + 1 1 2 6$  且 6 4 位元字料的最不重要一半，亦即  $D 0 - D 3$  被儲存在  $F P 2 i 1 2 8$ 。同時相反程序可被接著進行以將  $F P 2 i + 1 1 2 6$  及  $F P 2 i 1 2 8$  中的單精準度字元對儲存在記憶體 1 1 0， $F P 2 i + 1 1 2 6$  於位址 4 - 7， $F P 2 i 1 2 8$  於位址 0 - 3。

在另一特定實施例中 3 2 位元浮點暫存器可被分組為 4，亦即，四個一組。4 個單精準度字元可被載入及儲存自 / 至記憶體至 / 自該四單精準度暫存器。

## 五、發明說明 ( 7 )

在一特定實施例中載入及儲存指令其將資料轉換於一浮點暫存器及記憶體之間，可具有兩種定址模式：移位及索引。就移位定址而言，有效位址可被保有在一般目的暫存器中，而移位被給定為一立即值。立即值係由所存取資料之寬度而量計的。就索引定址而言，有效位址可藉由將一基準指標加以一索引而被計算出。基準指標及索引皆被保有於一般目的暫存器中，不像移位定址，索引無法藉由所存取之資料而被量計。

有三種不同的資料寬度可被用於載入及儲存作業：

一字首 `S` 指示將 32 位元資料轉換為一 4 位元組對準有效位址的一指令。浮點運算元  $FR_i$ ，其中  $i$  在  $[0, 63]$  之間，表示為 64 個單精準度浮點暫存器中之一。

一字首 `D` 指示將 64 位元資料轉換為一 8 位元組對準有效位址的一指令。浮點運算元  $DR_{2i}$ ，其中  $i$  在  $[0, 31]$  之間，表示為 32 個雙精準度浮點暫存器中之一。

一字首 `P` 指示將 32 位元資料值的向量轉換為一 8 位元對準有效位址的一指令。浮點運算元  $FP_{2i}$ ，其中  $i$  在  $[0, 31]$  之間，表示 32 對單精準度浮點暫存器中之一。

實施雙精準度及單精準度對的載入及儲存的指令的實例如表 1 所列。具有伴隨的 32 位元指令格式及算術的每一指令被列出以說明本發明的指令的特定實施例。

五、發明說明 ( 8 )

表 1

| 存取 | 模式 | 雙精準度        | 單精準度對       |
|----|----|-------------|-------------|
| 載入 | 索引 | F L D X . D | F L D X . P |
|    | 移位 | F L D . D   | F L D . P   |
| 儲存 | 索引 | F S T X . D | F S T X . P |
|    | 移位 | F S T . D   | F S T . P   |

F L D X . D 指令，表 2，使用暫存器加暫存器定址自記憶體載入一雙精準度浮點暫存器。有效位址藉由將 R<sub>m</sub> 加至 R<sub>n</sub> 而形成。由此有效位址所讀到的 64 位元被載入至 D R<sub>f</sub>。表 2 的頂列表示機器語言指令。以下部份為組合語言，接著為功能演算。

表 2

|                                                          |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |
|----------------------------------------------------------|--|----|----|--|----|------|--|----|----|--|----|---|--|---|---|---|--|
| 000101                                                   |  |    | m  |  |    | 0011 |  |    | n  |  |    | f |  |   | r |   |  |
| 31                                                       |  | 26 | 25 |  | 20 | 19   |  | 16 | 15 |  | 10 | 9 |  | 4 | 3 | 0 |  |
| FLDX.D Rm, Rn, DRf                                       |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |
| base $\leftarrow$ ZeroExtend64(Rm);                      |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |
| index $\leftarrow$ SignExtend64(Rn);                     |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |
| address $\leftarrow$ ZeroExtend64(base + index);         |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |
| result $\leftarrow$ FloatValue64(ReadMemory64(address)); |  |    |    |  |    |      |  |    |    |  |    |   |  |   |   |   |  |

F L D X . P 指令，表 3，使用暫存器加暫存器定址自記憶體載入一對單精準度暫存器。有效位址藉由將 R<sub>m</sub> 加至 R<sub>n</sub> 而形成。由此有效位址所讀到的 64 位元被載入至

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

五、發明說明 ( 9 )

F R <sub>i</sub> 。 6 4 位元的較低一半顯現在 F R <sub>i</sub> ，而較高一半在 F R <sub>i + 1</sub> 。表 3 以及以下所有表使用質表 2 相同的指令。

表 3

|                                       |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
|---------------------------------------|----|----|----|----|----|----|----|------|---|---|---|---|--|--|--|---|--|--|--|---|--|--|--|
| 000101                                |    |    |    | m  |    |    |    | 0111 |   |   |   | n |  |  |  | f |  |  |  | r |  |  |  |
| 31                                    | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9    | 4 | 3 | 0 |   |  |  |  |   |  |  |  |   |  |  |  |
| FLDX.P Rm, Rn, FPf                    |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| base ← ZeroExtend64(Rm);              |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| index ← SignExtend64(Rn);             |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| address ← ZeroExtend64(base + index); |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| result ← ReadMemoryPair32(address);   |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |

F S T X . D 指令，表 4 ，使用暫存器加暫存器定址自記憶體載入一雙精準度暫存器。有效位址藉由將 R <sub>m</sub> 加至 R <sub>n</sub> 而形成。由此有效位址所讀到的 6 4 位元被載入至 F R <sub>i</sub> 。 D R <sub>z</sub> 的 6 4 位元值被寫入至此有效位址。

表 4

|                                       |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
|---------------------------------------|----|----|----|----|----|----|----|------|---|---|---|---|--|--|--|---|--|--|--|---|--|--|--|
| 001101                                |    |    |    | m  |    |    |    | 0011 |   |   |   | n |  |  |  | z |  |  |  | r |  |  |  |
| 31                                    | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9    | 4 | 3 | 0 |   |  |  |  |   |  |  |  |   |  |  |  |
| FSTX.D Rm, Rn, DRz                    |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| base ← ZeroExtend64(Rm);              |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| index ← SignExtend64(Rn);             |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| value ← FloatValue64(DRz);            |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| address ← ZeroExtend64(base + index); |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| WriteMemory64(address, value);        |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |

F S T X . P 指令，表 5 ，使用暫存器加暫存器定址自記憶體載入一對單精準度暫存器。有效位址藉由將 R <sub>m</sub> 加

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

五、發明說明 ( 10 )

至 R<sub>m</sub> 而形成。由此有效位址所讀到的 64 位元被載入至 F R<sub>r</sub>。在 F P<sub>z</sub> 中的 64 位元資料被寫入此有效位址。64 位元資料的較低一半係來自 F R<sub>z</sub>，較高一半來自 F R<sub>z + 1</sub>。

表 5

| 001101                                |    |    |    | m  |    |    |    | 0111 |   |   |   | n |  |  |  | z |  |  |  | r |  |  |  |
|---------------------------------------|----|----|----|----|----|----|----|------|---|---|---|---|--|--|--|---|--|--|--|---|--|--|--|
| 31                                    | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9    | 4 | 3 | 0 |   |  |  |  |   |  |  |  |   |  |  |  |
| FSTX.P Rm, Rn, FPz                    |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| base ← ZeroExtend64(Rm);              |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| index ← SignExtend64(Rn);             |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| value ← FloatValuePair32(FPz);        |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| address ← ZeroExtend64(base + index); |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |
| WriteMemoryPair32(address, value);    |    |    |    |    |    |    |    |      |   |   |   |   |  |  |  |   |  |  |  |   |  |  |  |

F L D . D 指令，表 6，使用暫存器加上經量計之立即位址自記憶體載入一雙精準度浮點暫存器。有效位址藉由將 S 的標示展開 10 位元值乘以 8，並將之加至 R<sub>m</sub> 而形成。讀自此有效位址的 64 位元被載入 D R<sub>r</sub>。

表 6

| 100111                                        |    |    |    | m  |    |    |    | s |   |   |   | f |  |  |  | r |  |  |  |
|-----------------------------------------------|----|----|----|----|----|----|----|---|---|---|---|---|--|--|--|---|--|--|--|
| 31                                            | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9 | 4 | 3 | 0 |   |  |  |  |   |  |  |  |
| FLD.D Rm, s, DRf                              |    |    |    |    |    |    |    |   |   |   |   |   |  |  |  |   |  |  |  |
| base ← ZeroExtend64(Rm);                      |    |    |    |    |    |    |    |   |   |   |   |   |  |  |  |   |  |  |  |
| offset ← SignExtend10(s) << 3;                |    |    |    |    |    |    |    |   |   |   |   |   |  |  |  |   |  |  |  |
| address ← ZeroExtend64(base + offset);        |    |    |    |    |    |    |    |   |   |   |   |   |  |  |  |   |  |  |  |
| result ← FloatValue64(ReadMemory64(address)); |    |    |    |    |    |    |    |   |   |   |   |   |  |  |  |   |  |  |  |

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

五、發明說明 ( 11)

F L D . P 指令，表 7，使用暫存器加上經量計之立即位址自記憶體載入一單精準度浮點暫存器。有效位址藉由將 S 的標示展開 1 0 位元值乘以 8，並將之加至 R<sub>m</sub>而形成。讀自此有效位址的 6 4 位元被載入 F P<sub>f</sub>。6 4 位元資料的較低一半顯示在 F R<sub>f</sub>，而較高一半在 F R<sub>f + 1</sub>。

表 7

|                                        |    |    |    |    |    |    |    |   |   |   |   |   |  |
|----------------------------------------|----|----|----|----|----|----|----|---|---|---|---|---|--|
| 100110                                 |    | m  |    |    |    | s  |    |   |   | f |   | r |  |
| 31                                     | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9 | 4 | 3 | 0 |   |  |
| FLD.P Rm, s, FPf                       |    |    |    |    |    |    |    |   |   |   |   |   |  |
| base ← ZeroExtend64(Rm);               |    |    |    |    |    |    |    |   |   |   |   |   |  |
| offset ← SignExtend10(s) << 3;         |    |    |    |    |    |    |    |   |   |   |   |   |  |
| address ← ZeroExtend64(base + offset); |    |    |    |    |    |    |    |   |   |   |   |   |  |
| result ← ReadMemoryPair32(address);    |    |    |    |    |    |    |    |   |   |   |   |   |  |

F S T . D 指令，表 8，使用暫存器加上經量計之立即位址自記憶體載入一雙精準度浮點暫存器。有效位址藉由將 S 的標示展開 1 0 位元值乘以 8，並將之加至 R<sub>m</sub>而形成。D R<sub>z</sub> 的 6 4 位元值被寫入至此有效位址。

表 8

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

經濟部智慧財產局員工消費合作社印製

五、發明說明 ( 12 )

|                                                                                                                                                                      |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|----|----|----|----|----|---|---|---|---|--|---|--|--|--|
| 101111                                                                                                                                                               | m  |    |    |    | s  |    |    |   | z |   |   |  | r |  |  |  |
| 31                                                                                                                                                                   | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9 | 4 | 3 | 0 |  |   |  |  |  |
| FST.D Rm, s, DRz                                                                                                                                                     |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |
| base ← ZeroExtend64(Rm);<br>offset ← SignExtend10(s) << 3;<br>value ← FloatValue64(DRz);<br>address ← ZeroExtend64(base + offset);<br>WriteMemory64(address, value); |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |

F S T . P 指令，表 9，使用暫存器加上經量計之立即位址自記憶體載入一雙精準度浮點暫存器。有效位址藉由將 S 的標示展開 1 0 位元值乘以 8，並將之加至 R<sub>m</sub>而形成。F P<sub>z</sub> 的 6 4 位元值被寫入至此有效位址。6 4 位元資料的較低一半來自 F R<sub>z</sub>，而較高一半來自 F R<sub>z + 1</sub>。

表 9

|                                                                                                                                                                              |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|----|----|----|----|----|---|---|---|---|--|---|--|--|--|
| 101110                                                                                                                                                                       | m  |    |    |    | s  |    |    |   | z |   |   |  | r |  |  |  |
| 31                                                                                                                                                                           | 26 | 25 | 20 | 19 | 16 | 15 | 10 | 9 | 4 | 3 | 0 |  |   |  |  |  |
| FST.P Rm, s, FPz                                                                                                                                                             |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |
| base ← ZeroExtend64(Rm);<br>offset ← SignExtend10(s) << 3;<br>value ← FloatValuePair32(FPz);<br>address ← ZeroExtend64(base + offset);<br>WriteMemoryPair32(address, value); |    |    |    |    |    |    |    |   |   |   |   |  |   |  |  |  |

圖 3 顯示本發明的兩個浮點暫存器的內容的邏輯比較的布林結果。圖 3 顯示在 F P U 1 4 0 中的浮點暫存器，F P 暫存器 1，1 4 2 及 F P 暫存器 2，1 4 4 的兩個例子。例如，這些暫存器可為 3 2 或 6 4 位元長，而即，其對應上述的 F P 2<sub>i</sub> 或 D R<sub>i</sub>。比較運算 1 4 6 在兩個暫

(請先閱讀背面之注意事項再填寫本頁)

訂

線

五、發明說明 ( 13 )

存器 1 4 2 , 1 4 4 上實施。比較指令的例子如表 1 0 所列。例如, 結果 1 5 0 為一布林值, 真 / 假或為 1 或 0 的位元。結果 1 5 0 被送至 C P U 2 0 1 中的 L S U 2 4 0 ( 亦可見圖 5 , 其中顯示進一步細節 ) , 其接著將, 例如, 位元值送至一一般目的暫存器 ( G P R ) 2 7 1 , 其為 G P R 2 7 0 庫 ( 圖 5 ) 的一部份。G P R 可為一整數暫存器。

表 1 0

| 指令                    | 概要                                 |
|-----------------------|------------------------------------|
| FCMPEQ.S 來源 1,來源 2,結果 | 比較單精準度數求均等                         |
| FCMPEQ.D 來源 1,來源 2,結果 | 比較雙精準度數求均等                         |
| FCMPGT.S 來源 1,來源 2,結果 | 比較單精準度數求較大                         |
| FCMPGT.D 來源 1,來源 2,結果 | 比較雙精準度數求較大                         |
| FCMPGE.S 來源 1,來源 2,結果 | 比較單精準度數求大於等於                       |
| FCMPGE.D 來源 1,來源 2,結果 | 比較雙精準度數求大於等於                       |
| FCMPGE.S 來源 1,來源 2,結果 | 比較單精準度數求不規則, 例如, X 及 Y 的至少之一為一 NaN |
| FCMPUN.D 來源 1,來源 2,結果 | 比較雙精準度數求不規則                        |

表 1 1 顯示由在表 1 0 中之以上浮點指令映射至 I E E E 7 5 4 比較功能之結果。在表 1 1 中, 比較的結果可被以一布林函數儲存在一一般目的暫存器 ( G P R

五、發明說明 ( 14 )

) R<sub>i</sub> 中，其中 G P R 可為一整數暫存器。

表 1 1

| 格 式 | 指 令                                                       | IEEE 754 比 較                      |
|-----|-----------------------------------------------------------|-----------------------------------|
| 單   | FCMPEQ.S FR <sub>0</sub> ,FR <sub>1</sub> ,R <sub>d</sub> | FR <sub>0</sub> =FR <sub>1</sub>  |
|     | FCMPGT.S FR <sub>0</sub> ,FR <sub>1</sub> ,R <sub>d</sub> | FR <sub>0</sub> >FR <sub>1</sub>  |
|     | FCMPGE.S FR <sub>0</sub> ,FR <sub>1</sub> ,R <sub>d</sub> | FR <sub>0</sub> >=FR <sub>1</sub> |
|     | FCMPUN.S FR <sub>0</sub> ,FR <sub>1</sub> ,R <sub>d</sub> | FR <sub>0</sub> ?FR <sub>1</sub>  |
| 雙   | FCMPEQ.D DR <sub>0</sub> ,DR <sub>2</sub> ,R <sub>d</sub> | DR <sub>0</sub> =DR <sub>2</sub>  |
|     | FCMPGT.D DR <sub>0</sub> ,DR <sub>2</sub> ,R <sub>d</sub> | DR <sub>0</sub> >DR <sub>2</sub>  |
|     | FCMPGE.D DR <sub>0</sub> ,DR <sub>2</sub> ,R <sub>d</sub> | DR <sub>0</sub> >=DR <sub>2</sub> |
|     | FCMPUN.D DR <sub>0</sub> ,DR <sub>2</sub> ,R <sub>d</sub> | DR <sub>0</sub> ?DR <sub>2</sub>  |

在一特定實施例中，以下兩實例指令，

F C M P G E . S 及 F C M P G E . D ，在表 1 2 及表 1 3 中說明浮點比較指令的格式。

F C M P G E . S 指令，表 1 2 ，實施一單精準度浮點大於或等於比較。若 F R<sub>g</sub> 大於或等於 F R<sub>h</sub> ，則設 R<sub>d</sub> 為 1 ，其它則設 R<sub>d</sub> 為 0 。

表 1 2

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

五、發明說明 ( 15 )

|                              |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
|------------------------------|----|----|----|----|------|----|----|---|---|---|---|--|--|---|--|--|--|
| 001100                       | g  |    |    |    | 1110 | h  |    |   |   | d |   |  |  | r |  |  |  |
| 31                           | 26 | 25 | 20 | 19 | 16   | 15 | 10 | 9 | 4 | 3 | 0 |  |  |   |  |  |  |
| FCMPGE.S FRg, FRh, Rd        |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| source1 ← FloatValue32(FRg); |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| source2 ← FloatValue32(FRh); |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| result ← source1 >= source2; |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| Rd ← Register(result);       |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |

F C M P G E . D 指令，表 1 3，實施一雙精準度浮點大於或等於比較。若 D R<sub>g</sub> 大於或等於 D R<sub>h</sub>，則設 R<sub>d</sub> 為 1，其它則設 R<sub>d</sub> 為 0。

表 1 3

|                              |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
|------------------------------|----|----|----|----|------|----|----|---|---|---|---|--|--|---|--|--|--|
| 001100                       | g  |    |    |    | 1111 | h  |    |   |   | d |   |  |  | r |  |  |  |
| 31                           | 26 | 25 | 20 | 19 | 16   | 15 | 10 | 9 | 4 | 3 | 0 |  |  |   |  |  |  |
| FCMPGE.D DRg, DRh, Rd        |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| source1 ← FloatValue64(DRg); |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| source2 ← FloatValue64(DRh); |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| result ← source1 >= source2; |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |
| Rd ← Register(result);       |    |    |    |    |      |    |    |   |   |   |   |  |  |   |  |  |  |

硬體實施之特定實施例

在一特定實施例中，本發明可在一具有一包括 6 個單元及一可拆式浮點單元 ( F P U ) 的一核心單元 2 0 0 的 C P U 中被實施。圖 4 說明 S 5 核心之頂級分隔的一例。表 1 4 敘述在 S 5 核心中的每一單位的功能。指令流單元 ( I F U ) 2 1 0，其處理浮點指令，在附件 3 中被進一

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

五、發明說明 ( 16 )

步敘述。在一特定實施例中 F P U 2 6 5 實施如 I E E E 7 5 4 浮點標準中所指明的位址運算，以硬體或軟體或兩者之組合。F P U 2 6 5 的另一特定實施例如附件 1 及 2 中所示。

表 1 4

| 單元          | 代號  | 敘述                               |
|-------------|-----|----------------------------------|
| S5 核心 200   | S5  | 頂層核心區塊                           |
| 滙流排界面單元 205 | BIU | 控制滙流排對外部模組之存取,例如週邊模組及外部記憶體界面     |
| 指令流單元 210   | IFU | CPU 管路的前端:抓取,結碼,發送及分支。亦包含模式 B 對抗 |
| 指令多媒體單元 220 | IMU | 處理所有整數及多媒體指令。主 CP U 資料路徑         |
| 指令快取單元 230  | ICU | 包含指令快取器及指令翻譯察看緩衝器(TLB)           |
| 載入儲存單元 240  | LSU | 處理所有記憶體指令及資料快取控制                 |
| 資料快取單元 250  | DCU | 包含資料快取及資料翻譯察看緩衝器(TLB)            |
| 浮點單元 265    | FPU | 可拆卸式浮點單元(在圖 4 中未顯示)              |

圖 5 說明在 C P U 中有限數量之所選定之區塊的一特

五、發明說明 ( 17 )

定實施例。在圖 5 中，C P U 2 0 1 包括區塊 I F U 2 1 0，I M U 2 2 0，及 L S U 2 2 0。F P U 2 6 5 經由一界面匯流排 2 6 0 將資料轉送至及自 L S U 2 4 0。L S U 接著可經由 Isu\_result\_ex3 匯流排 2 6 2 將資料傳送至 G P R 2 7 0。自 G P R 2 7 0 至 L S U 2 4 0 的返回路徑可能如虛線匯流排 2 6 2 所指示的是非直接的。在 L S U 2 4 0 自 G P R 2 7 0 取得資料後其接著將之傳送到 F P U 2 6 5。L S U 2 4 0，如圖 4 所示，亦將資料載入並儲存至記憶體。

F P U 可為具有階段 F 1 及 F 2 的管路。F P U 2 6 5 包括 6 4 個 3 2 位元暫存器，其可被用作單精準度浮點暫存器 ( F R 暫存器 )；各包含 3 2 位元，或雙精準度浮點暫存器 ( D R 暫存器 )，各包含 6 4 位元。自 D R 組至 F R 組的映射係如達成如下：D R 2 i 的下半，其中 i 為 [ 0，3 1 ] 之間，映射至 F R 2 i；且 D R 2 i 的上半，其中 i 為 [ 0，3 1 ] 之間，映射至 F R 2 i + 1。在一替代實施例中，自 D R 組至 F R 組至 F R 組的映射係完成如下：D R 2 i 的上半，其中 i 為 [ 0，3 1 ] 之間，映射至 F R 2 i；且 D R 2 i 的下半，其中 i 為 [ 0，3 1 ] 之間，映射至 F R 2 i + 1。再者，F R 組可被成雙存取，如向量：有 3 2 × 2 元素對稱為 F R 2 i，其中 i 為 [ 0，3 1 ] 之間，且每一對包含浮點暫存器 F R 2 i 及 F R 2 i + 1。F P U 2 6 5 亦包括一 F P S C R，其包括浮點狀態及控制暫存器。此主要係用

五、發明說明 ( 18 )

以控制浮點演算的近似模式及例外行為。F P S C R 為 3 2 位元寬。

在一特定實施例中 L S U 2 4 0 實施至及自 F P U 2 6 5 的載入及儲存，包括接收 6 4 位元浮點資料，接收一位元浮點比較結果，及將結果傳送至一般目的暫存器 ( G P R ) 。對一 F P U 而言載入 L S U 2 4 0 可自記憶體執行一典型載入作業。在資料已被載入然後結果可從 L S U 中之對準器被送至 F P U 。對一 F P U 而言儲存 L S U 係一對記憶體的一般儲存。差別為在 E 2 階段，亦即 I F U 管路的第二階段，來自 F P U 的儲存資料將取代從 I F U 2 1 0 傳來的 i f u - s r c 3 ( 圖 4 ) 資料。為供 F P U 2 6 5 傳送至 G P R ，資料自 E 2 中的 F P U 接收且可經由對準器推至 Isu\_result\_ex3 ( 圖 4 ) 而至 G P R 或其可被轉傳。F P U 2 6 5 比較結果，其可為一單一位元，可能無法用於傳送至一 G P R ，直到 F P U 2 6 5 的第二管路階段，F 2 ，結束。如此信號可在 E 3 中被用到，亦即，I F U 2 1 0 的第三管路階段，以控制在 L S U 2 4 0 中的對準器的結果的底部位元組的混合，使比較結果可被放在 Isu\_result\_ex3 至 G P R 。

結 論

在前述的說明書中，本發明已參照其特定實例實施例而被敘述。其它實施例對於熟習該項技術者是很清楚的，例如，指令可為 1 6 或 6 4 或 1 2 8 位元的長度，且微處

## 五、發明說明 ( 19 )

理器及／或副處理器可以 1 6 , 3 2 或 1 2 8 位元之匯流排或字元為基礎進行運作。如此很明顯的在不脫離本案之申請專利範圍所定義的本發明的較廣精神及範圍之下，可有許多不同的變更及改變。

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

四、中文發明摘要 (發明之名稱： 浮點指令組結構及其應用方法 )

處理器及副處理器元件，構建以如所述地執行一浮點指令組結構。舉例說明一浮點副處理器及記憶體之間的雙精準度字元及／或單精準度字元對的載入及儲存，以及在一CPU中之一整數暫存器中之一浮點單元中的兩個浮點數之布林比較結果的儲存。

英文發明摘要 (發明之名稱： A FLOATING POINT INSTRUCTION SET ARCHITECTURE AND IMPLEMENTATION )

Processor and co-processor elements, structured to execute a floating-point instruction set architecture, as described. Examples are given of loading and storing double precision words and/or pairs of single precision words between a floating-point co-processor and memory, as well as storing the Boolean result of comparing two floating-point numbers in a floating-point unit in an integer register in a CPU.

經濟部智慧財產局員工消費合作社印製

(請先閱讀背面之注意事項再填寫本頁各欄)

裝

訂

## 六、申請專利範圍

1 . 一種用於自一記憶體載入資料至一處理器的方法，包含：

由一電腦指令中之一作業碼決定一載入作業；

依照該作業碼自一 K 位元寬記憶體 N 群組資料載入至與該處理器相關的一儲存位置，其中該群組包含 M 位元組，且 K，M 及 N 為正整數。

2 . 如申請專利範圍第 1 項的方法，其中該儲存位置包含一 64 位元暫存器。

3 . 如申請專利範圍第 1 項的方法，其中該儲存位置包含一對 32 位元暫存器。

4 . 如申請專利範圍第 1 項的方法，其中 K 等於 32，N 等於 1 且 M 等於 8 位元組。

5 . 如申請專利範圍第 1 項的方法，其中 K 等於 32，N 等於 2 且 M 等於 4 位元組。

6 . 如申請專利範圍第 1 項的方法，其中 K 等於 32，N 等於 4 且 M 等於 4 位元組。

7 . 如申請專利範圍第 1 項的方法，其中該儲存位置包含至少 N 乘以 M 位元組之長度。

8 . 如申請專利範圍第 1 項的方法，其中 K 等於 64，N 等於 2，M 等於 8 位元組。

9 . 一種用於自一記憶體儲存資料至一處理器的方法，包含：

由電腦指令中之一作業碼決定一儲存作業；

依照該作業碼自與該處理器相關的 N 個儲存位置儲存

(請先閱讀背面之注意事項再填寫本頁)

訂

線

## 六、申請專利範圍

一 K 位元寬之資料，其中每一儲存位置包含 M 位元組，且 K，M，及 N 為整數。

1 0 . 如申請專利範圍第 9 項的方法，其中該儲存位置係包含一 6 4 位元暫存器及一對 3 2 暫存器所組成之一群組中所選出。

1 1 . 如申請專利範圍第 9 項的方法，其中 K 等於 3 2，N 等於 1 及 M 等於 8 位元組。

1 2 . 如申請專利範圍第 9 項的方法，其中 K 等於 3 2，N 等於 2 及 M 等於 4 位元組。

1 3 . 如申請專利範圍第 9 項的方法，其中 K 等於 3 2，N 等於 4 及 M 等於 4 位元組。

1 4 . 一種自一記憶體載入資料至一處理器的方法，包含：

由一電腦指令中之一作業碼決定一載入作業；

依照該作業碼由一包含一雙精準度資料字元的一單精準度寬記憶體載入至與該處理器相關的兩個單精準度暫存器。

1 5 . 一種用以自一記憶體將資料載入至一處理器的方法，包含：

由在電腦指令中之一作業碼決定一載入作業；

依照該作業碼由一包含兩個單精準度資料字元的單精準度寬記憶體載入至與該處理器相關的兩個單精準度暫存器。

1 6 . 一種自一記憶體將資料儲存至一處理器的方法

## 六、申請專利範圍

， 包含：

由在電腦指令中之一作業碼決定一儲存作業；

依照該作業碼自與該處理器相關的兩個單精準度暫存器儲存至包含一雙精準度資料字元的一單精準度寬記憶體。

1 7 . 一種自一記憶體將資料儲存至一處理器的方法

， 包含：

由在電腦指令中之一作業碼決定一儲存作業；

依照該作業碼自與該處理器相關的兩個單精準度暫存器儲存至包含一單精準度資料字元的一單精準度寬記憶體。

1 8 . 一種在一電腦系統中比較多數浮點數的方法，

包含：

由一電腦指令中之一作業碼決定一比較作業；

將該多數浮點數之一第一浮點數與該多數浮點數之一第二浮點數作比較以取得一布林結果；

將該布林結果儲存於一儲存位置，其中該儲存位置為一一般目的暫存器（G P R）。

1 9 . 如申請專利範圍第 1 8 項的方法，進一步包含使用該結果於一條件分支指令。

2 0 . 如申請專利範圍第 1 8 項的方法，其中該 G P R 包含一整數暫存器。

2 1 . 如申請專利範圍第 1 8 項的方法，其中該布林結果為由包含位元 I 及 O 的群組中所選出。

## 六、申請專利範圍

2 2 . 一種在一電腦系統中比較多數浮點數的方法，  
包含：

由一電腦指令中之一作業碼決定一比較作業；

將該多數浮點數之一第一浮點數與該多數浮點數之一  
第二浮點數作比較以取得一布林結果；

將該布林結果儲存於一儲存位置，其中該儲存位置為  
一判定暫存器，以使若其為真，則指令被執行，而若為假  
，則指令被以一 N O P （不執行指令）處理。

（請先閱讀背面之注意事項再填寫本頁）

訂

線

89120417

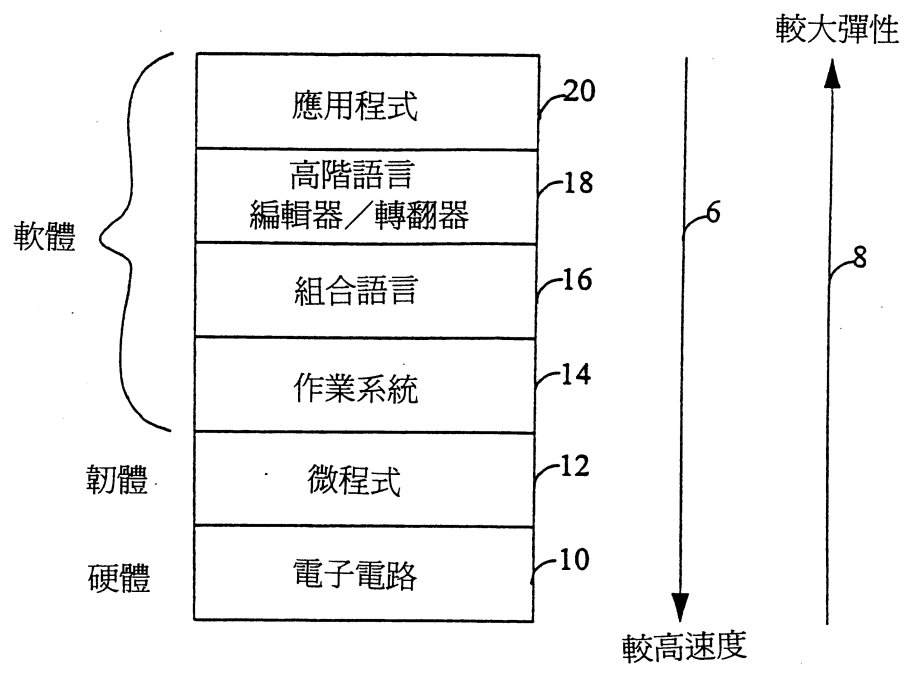


圖 1

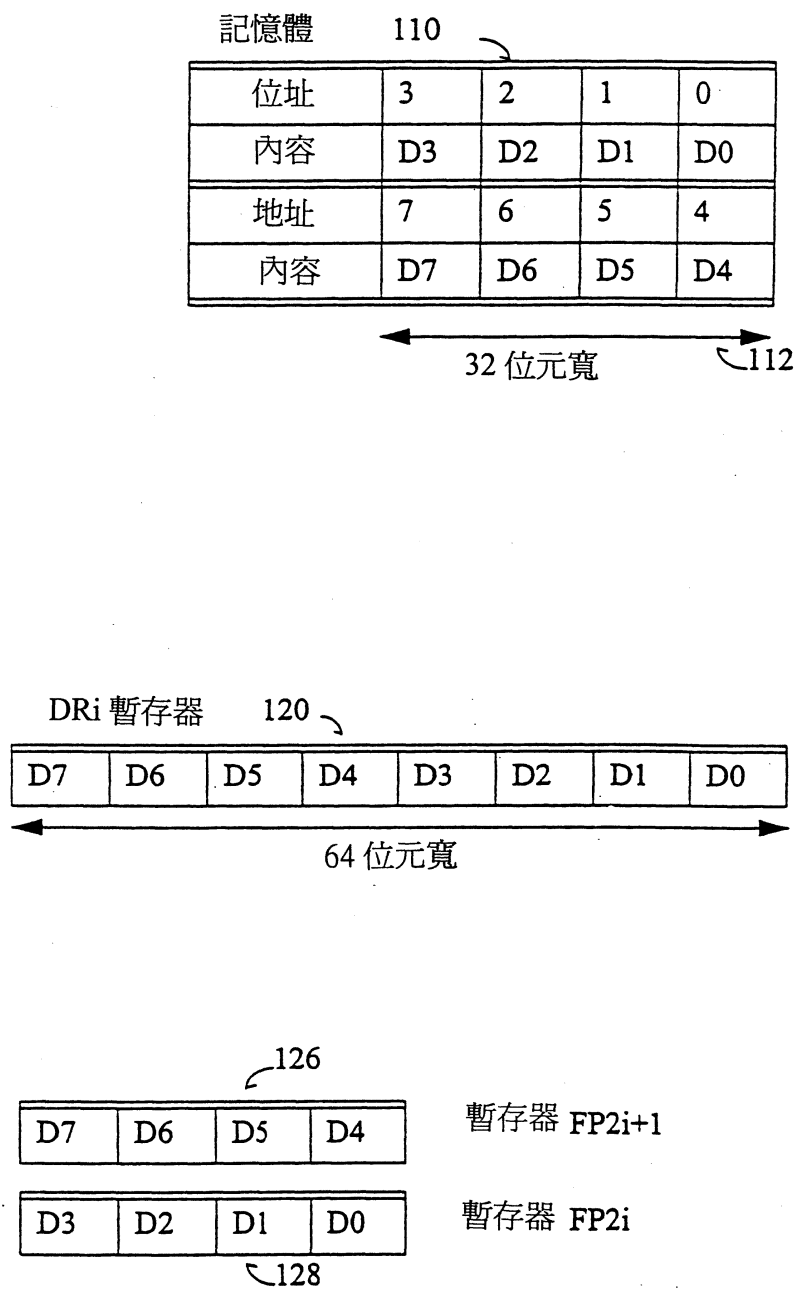


圖 2

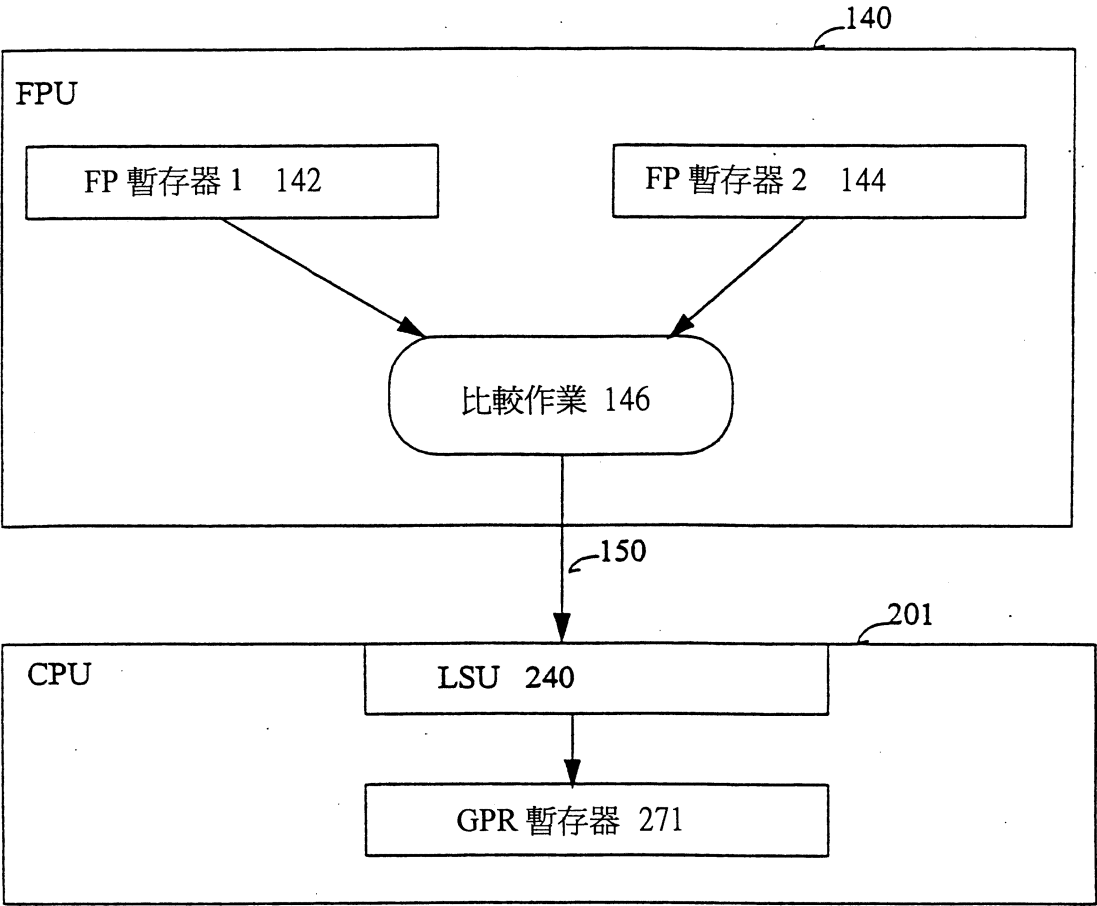


圖 3

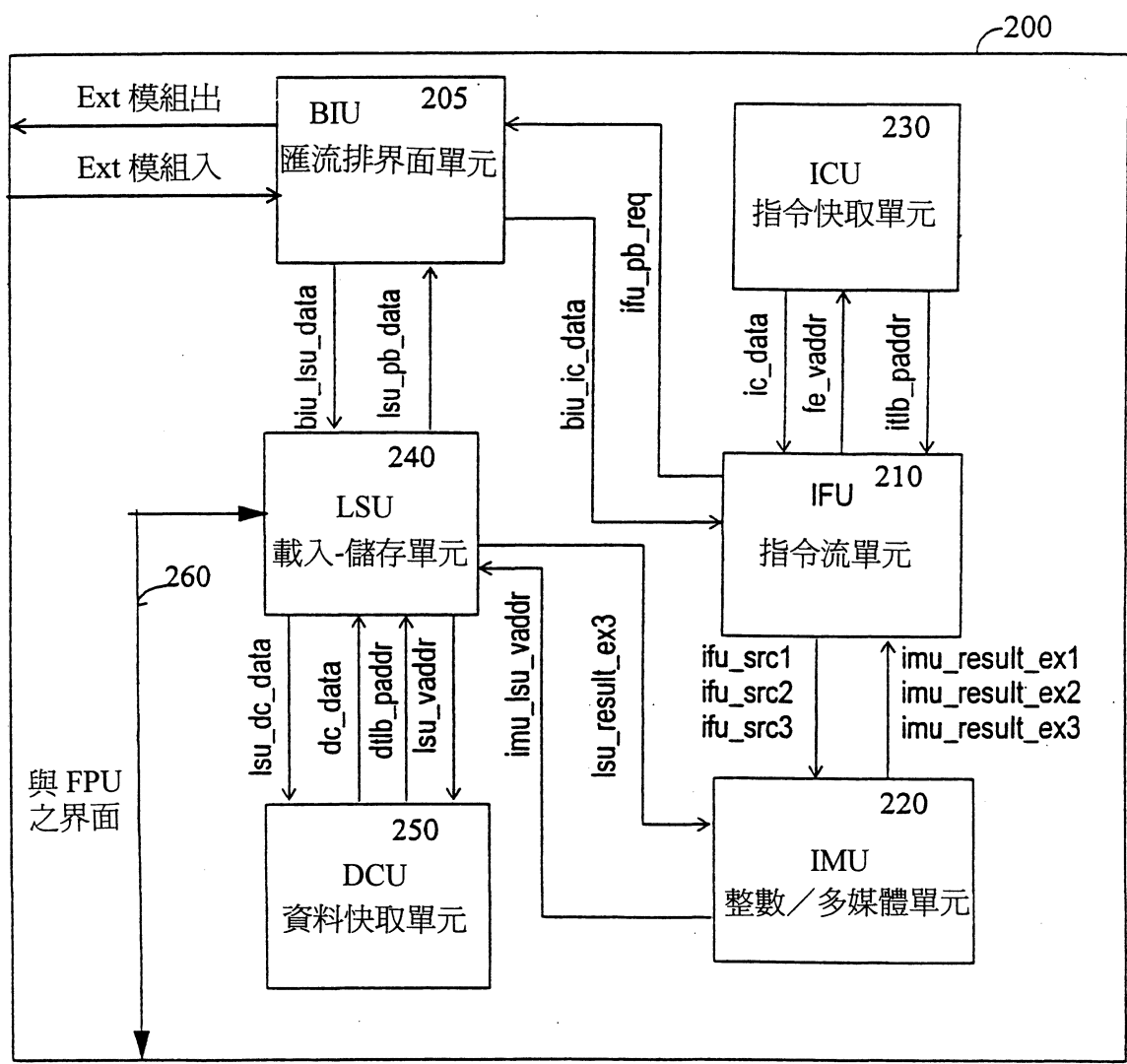


圖 4

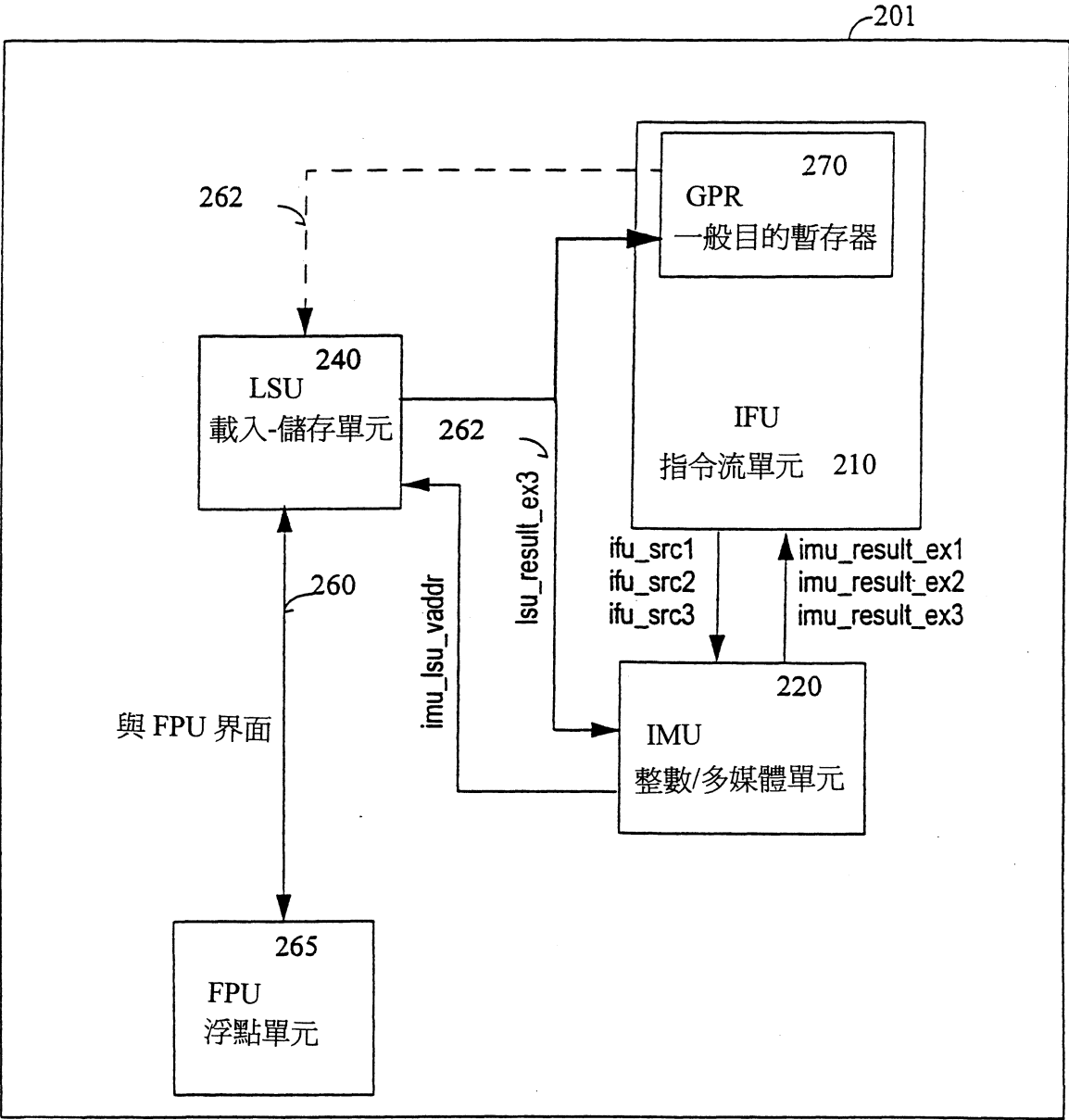


圖 5