

公告

申請日期	87.6.8
案 號	87109037
類 別	G06F 7/544

A4
C4

434507

(以上各欄由本局填註)

發明專利說明書

一、發明 名稱	中 文	基本函數之高準確度預估
	英 文	"HIGH ACCURACY ESTIMATES OF ELEMENTARY FUNCTIONS"
二、發明 人	姓 名	1.馬汀 史丹利 史奇摩勒 2.都諾 諾曼 先濟
	國 籍	均美國
	住、居所	1.美國德州奧斯汀市石角大道7504號 2.美國德州奧斯汀市因伯格蘭大道10428號
三、申請人	姓 名 (名稱)	美商萬國商業機器公司
	國 籍	美國
	住、居所 (事務所)	美國紐約州阿蒙市
	代 表 人 姓 名	費羅普

裝

訂

線

434507

(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

1998年2月2日

09/017,233

☒有 ☐無主張優先權

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

經濟部中央標準局員工消費合作社印製

五、發明說明(1)

發明背景

1. 發明領域

本發明概言之係關於電腦系統，且更明確地說係關於一種可提供一改良方式法之電腦系統，且該改良式方法是用以產生平方根，平方根倒數，及倒數函數之高準確度估計。

2. 相關技術說明

電腦處理器可對不同型式之數字或運算元執行算術運算。例如，最簡單之運算涉及整數運算元，而整數運算元是利用"定點"表示法來加以表示。非整數一般是依照"浮點"表示法來加以表示。在傳統之電腦中，一"單準度"浮點數字是以利用32-位元(一字組)之欄位來加以表示，而一"倍準度"浮點數字則是利用一64-位元(二字組)之欄位來加以表示。

浮點表示法(其也稱為指數表示法)可用以表示非常大及非常小之數字。浮點表示法具有三部份：一假數(或有效數字)，一指數，及一正負號(正或負)。假數指定數字之數元，而指數指定數字之大小，亦即基數之次方，而該基數之次方要與假數相乘以產生該數字。例如，如果使用基數10，則數字28330000可表示成為2833E+4(假數為2833且指數為4)，而數字0.054565可表示成為54565E-6(假數為54565且指數為-6)。因為處理器使用二進位值於他們之計算，所以大多數電腦之浮點數字使用2做為基數(數基)。因此，浮點數字通常可利用二進位項依照下列型態來加以

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明(2)

表示

$$n=(-1)^S \cdot 1.F \cdot 2^E,$$

其中 n 是浮點數字(基數為 10)， S 是數字之正負號(如果數字為正則 $S=0$ ，否則 $S=1$)， F 假數之小數部份(基數為 2)，而 E 是數基之指數。根據電機電子工程師協會(IEEE)標準 754，單精度浮點數字如下使用 32 位元：第一位元顯示正負號(S)，接著八位元顯示偏移量為 127 之指數(E +偏移量)，且最後 23 位元顯示小數(F)。所以，例如，十進位數字 10 可表示成為下列 32-位元值：

0 10000010 010000000000000000000000

因為此數字對應於 $(-1)^0 \cdot 1.01_2 \cdot 2^{130-127} = 1.25 \cdot 2^3 = 10$ 。

許多處理器利用浮點單元(FPU)來處理浮點運算。浮點處理可用於加法，乘法及除法運算，且也可用於其他特殊數學函數，例如平方根($\sqrt{x}$)，平方根倒數($1/\sqrt{x}$)，及倒數($1/x$)函數。許多傳統處理器使用疊代型近似演算法來決定該等值，例如牛頓-瑞佛森(Newton-Raphson)疊代法。若要瞭解此種特殊演算法，首先瞭解用以尋找零根之牛頓疊代法會有幫助，而牛頓疊代法也可用於除法運算，且是展示於圖 1。一廣義曲線是以函數 $f(x)$ 來表示，且想要知道此函數之零根(亦即 x 截距之值)。基本上，此函數是以其在任一任意位置(x_0)之切線來加以近似，且 x_0 對應於啓始估測，且新估測接著是以此切線之零根(x_1)為基礎來公式化求

五、發明說明(3)

得。此種簡單程序接著可重複達成所要準確度所需之任意次數。一般而言，可以證實對應於一給定疊代*i*之切線之零根可由下列方程式求得：

$$X_{i+1} = X_i - \frac{f(X_i)}{f'(X_i)} \quad (\text{方程式 1})$$

此倒數運算可藉由考慮函數 $f(x)=1/x-r$ 來塑造成為一尋找零根運算。因為此函數之零根是 $1/r$ ，所以牛頓疊代法可用以計算此倒數函數。可以證實，對於此倒數函數而言，方程式1變為

$$X_{i+1} = X_i(2 - X_i r) \quad (\text{方程式 2})$$

此方程式常用於除法。 X_0 之啓始值是由一檢查表來提供。使用方程式2之許多疊代來求得對應於*r*之除數之倒數。該倒數值接著乘以被除數以求得商數。

若要計算一平方根，可執行類似之疊代程序，且可以證實平方根可依照下列方程式來加以計算：

$$X_{i+1} = 0.5 (X_i + r/X_i) \quad (\text{方程式 3})$$

但是，方程式3需要進行一相除運算， r/X_i ，而相除運算遠慢於相乘運算。相對地，用以計算平方根倒數之牛頓-瑞佛森疊代法經常受到使用，其中：

$$X_{i+1} = 0.5 X_i (3 - X_i^2 r) \quad (\text{方程式 4})$$

若要求得*r*之平方根倒數之較準確估計可進行許多次疊

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明(4)

代。此值接著可乘以 r 來求得 r 之平方根，如下所示：

$$\sqrt{r} = r(1/\sqrt{r})。$$

牛頓-瑞佛森疊代法以平方型速率來收斂，所以每一疊代大約加倍正確數元之數目。再一次該程序是使用檢查表來加以啓始，而此可跳過最先一些疊代且可改善效能。若要藉由檢查表來產生四正確位元，則需要一8-輸入項之表。所需之位元每增加一位元會使表輸入項之數目加倍。

牛頓-瑞佛森疊代演算法應用於許多情形，因為他們通常快於其他演算法。但是，疊代演算法可能造成其他問題。例如，其可導致捨入誤差，且對於除法運算而言，其未提供餘數。另外，疊代演算法仍具有效能限制。決定於所需之準確度，完整之疊代程序可需要相當多之處理器循環。對應之延遲可能會影響某些程序，尤其是涉及多媒體展示之程序。處理器架構之某些多媒體延伸也指定一些倒數及平方根倒數指令，且該等指令需要提高之準確度(12位元)。若要使用傳統之檢查表技術來產生12正確位元，則需要一具有2048輸入項之表，而該表意謂更高之硬體成本。一替代方案是使用一具有64輸入項之表與一牛頓-瑞佛森疊代，且該牛頓-瑞佛森疊代需要至少二相乘運算，如方程式2所示(用於倒數)，或需要至少三相乘運算，如方程式4所示(用於平方根倒數)。

另一替代方案涉及使用一線性近似。請參看，例如，美國專利第5,563,818號，與"Efficient Initial Approximation

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

以

五、發明說明(5)

and Fast Converging Methods for Division and Square Root," Proceedings 12th Symposium on Computer Arithmetic, 第2-9頁(1995)。但是,該等建構所需之檢查表及乘法器大於下文將會說明之建構。一執行平方近似之系統展示於美國專利第5,245,564號。此種技術是線性近似之延伸,且對於相同數目之字組而言,可提供較檢查表高三倍之準確度,而所需之代價是一額外之相乘及相加,但是因此需要較長之執行時間。

藉由前文所述,最好能夠設計一種改良式方法,且該種方法可準確地估計平方根,平方根倒數,及倒數,以減少疊代之數目。如果該方法可降低乘法器之大小及延遲以達成較高之準確度,而無需增加計算倒數或平方根倒數所需之時間,將更具優勢。

發明之簡要說明

因此本發明之一目標是提供一種方法,且該種方法利用電腦系統之處理器來計算一數字之平方根,平方根倒數,及倒數。

本發明之另一目標是提供一此種方法,且該種方法可導致高準確度之估計,但無需極多之疊代步驟。

本發明之另一目標是提供也無需大幅增加硬體(亦即使使用較小之檢查表)之一此種方法。

前述目標可利用本發明之第一建構來達成,且該第一建構是調整成為利用一具有處理器之電腦系統來提供一輸入值之倒數。該第一建構通常包含下列步驟:正規化該輸入

五、發明說明(6)

值，以經正規化之輸入值為基礎自一表選擇第一常數及第二常數，其中第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/x$ 的一線性近似為基礎，且受到偏移來降低該給定間段之最大誤差值，與藉由自第二常數減去經正規化之輸入值及第一常數之乘積來計算一估計之倒數。經正規化之輸入值具有一假數，且第一及第二常數是以該假數之最高有效位元為基礎來加以選擇。在一變型中，輸入值是在範圍 $1 \leq x < 2$ 之中受到正規化，且該表之間段大小是 $1/32$ 。第一及第二常數最好是以足夠之準確度來提供以致估計之倒數之相對誤差小於 $1/4096$ 。在另一較佳實例中，第二常數是一給定間段之左及右邊緣之值的函數，且只有假數之低階部份受到使用，因而減少相乘-相加運算所需之位元數目。因而在硬體成本相當小幅成長之下，計算一數字之倒數所需之時間可獲得重大改善。

在本發明之第二建構中，且該第二建構是調整成為利用一具有處理器之電腦系統來提供一輸入值之平方根倒數，該方法通常包含下列類似步驟：正規化輸入值，以經正規化之輸入值為基礎自一表選擇第一常數及第二常數，其中第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/\sqrt{x}$ 的一線性近似為基礎，且受到偏移來降低該給定間段之最大誤差值，與藉由自第二常數減去經正規化之輸入值及第一常數之乘積來計算一估計之平方根倒數。在此建構之一變型中，輸入值是在範圍 $0.5 \leq x < 2$ 之中受到正規化，且二檢查表受到使用，而在 $0.5 \leq x < 1$ 之範圍中第一表之間

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明(7)

段大小是 $1/32$ ，且在 $1 \leq x < 2$ 之範圍中第二表之間段大小是 $1/16$ 。第一及第二常數最好是以足夠之準確度來提供以致估計之平方根倒數之相對誤差小於 $1/4096$ 。在另一較佳實例中，第二常數是一給定間段之左及右邊緣之值的函數，且只有假數之低階部份受到使用，因而減少相乘-相加運算所需之位元數目(假數之高階部份是用以自該二表來選擇該二常數)。一估計之平方根可藉由相乘經正規化之輸入值及估計之平方根倒數來加以計算。再一次，在硬體成本相當小幅成長之下，計算一數字之平方根倒數所需之時間可獲得重大改善。

藉由下列詳細之書面說明應可明瞭本發明之前述以及其他目標，特點，及優點。

圖式之簡要說明

據信為本發明之特徵之新奇特點陳述於附加之申請專利範圍。但是，當連同附圖來閱讀一示範實例之下列詳細說明時，應可最容易瞭解本發明本身，以及本發明之一較佳使用模式，其他目標，及優點，其中：

圖1是用以展示牛頓尋找零根法之圖形，且牛頓尋找零根法構成用以決定平方根，平方根倒數，或倒數之傳統疊代法的基礎；

圖2是一處理器之實例之方塊圖，且本發明可建構於該處理器；

圖3是展示本發明之方法之圖形，而該方法可利用非疊代方式來求得平方根，平方根倒數，及倒數之高準確度估

五、發明說明(8)

計：

圖4是本發明之一實例之方塊圖；且

圖5是本發明之另一實例之方塊圖，且該實例共享一傳統浮點單元之乘法器之一些電路。

較佳實施例說明

本發明係針對一種用以決定一數字之平方根，平方根倒數，或倒數之方法，且該方法是由一電腦系統之處理器來執行。請參看圖2，一處理器之方塊圖受到展示，其中可建構本發明之一較佳實例。該處理器可包含圖2所示之各種元件，但該處理器無需是傳統型，亦即其也可包含新硬體元件，或具有現有元件之新奇連接架構。因此，雖然本發明可藉由參考圖2來加以瞭解，但是此種參考不應視為一種限制。

處理器100包含匯流排介面單元102，而匯流排介面單元102控制處理器100與資料處理系統之剩餘部份(未受到展示)間之資料流動。匯流排介面單元102連接至資料快取記憶體104與指令快取記憶體106。指令快取記憶體106提供指令給分支單元108，而分支單元108決定何種序列之指令適合處理器100之通用暫存器(GPRs)110及浮點暫存器(FPRs)112之內容，載入/儲存單元114，定點執行單元116，及浮點執行單元118之可用性，與該等指令本身之本質。分支單元108轉送經排序之指令至分派單元120，而分派單元120發出個別之指令至適當之邏輯單元(載入/儲存單元114，定點執行單元116，浮點執行單元118)。

五、發明說明(9)

定點執行單元116自通用暫存器110讀取資料並寫入資料至通用暫存器110。浮點執行單元118自浮點暫存器112讀取資料並寫入資料至浮點暫存器112。載入/儲存單元114自通用暫存器110，浮點暫存器112讀取資料，且寫入資料至資料快取記憶體104或外部記憶體(未受到展示)，決定於資料處理系統所採用之記憶體階層及快取協定，而此超出本發明之範圍。載入/儲存單元114也自資料快取記憶體104讀取資料，且寫入資料通用暫存器110及浮點暫存器112。

熟悉本技術領域者應可理解處理器100之建造或其之運算的細節皆可隨著影響設計之目標而改變。例如，處理器100可包含暫存器重新命名多重定點執行單元116以並行執行無資料相依性之定點指令，或一記憶體管理單元以調節資料快取記憶體104及處理器100以外之第二階(L2)快取記憶體(未受到展示)之內容。浮點執行單元118可為接收三輸入運算元(A，B及C)之型式，且該三輸入運算元是表示成為浮點數字。浮點執行單元118使用該等運算元來執行"相乘-相加"指令。相乘-相加指令執行算術運算 $\pm[(A \times C) \pm B]$ 。運算元A，B，及C之指數部份提供給一指數計算器，運算元A及C之假數部份提供給一乘法器，而運算元B之假數部份提供給一校準移位器，而該校準移位器校準運算元B與A及C之乘積，以他們之個別指數為基礎。經校準之運算元B接著加至A乘以C之乘積。

浮點單元118可根據本發明之第一建構來使用以產生一

(請先閱讀背面之注意事項再填寫本頁)

表

訂

五、發明說明(10)

浮點數字之倒數之假數。自倒數至商數之步驟，及當計算倒數時之浮點指數之處理，為眾所知且在本文將不加以討論。浮點單元118也可根據本發明之第二建構來使用以產生一浮點數字之平方根倒數之假數。該平方根接著也可獲得決定，藉由相乘該平方根倒數及該數字本身。

藉由參看圖3可進一步瞭解本發明之方法，而圖3，根據本發明之第一建構，可視為倒數函數 $y=1/x$ 之一區段的圖形。假設 y 及 x 是經正規化之數字，且 $1 \leq x < 2$ 。如前所述，如何處理浮點指數為眾所知，所以在不失去一般性之下，本文將只討論假數之產生。在圖3中，曲線A-B表示函數 $y=1/x$ 之一區段。在二點 x_1 及 x_2 之間，線C-D近似 $1/x$ 。線C-D之斜率 a 可由下列方程式來求得：

$$a = (y_2 - y_1) / (x_2 - x_1) = (1/x_2 - 1/x_1) / (x_2 - x_1) = -1 / (x_2 x_1)$$

偏移 b 則可由下列方程式來求得：

$$b = y_1 - ax_1$$

發生於 $x_{\max}$ 之最大相對誤差 y_{err} 可藉由下列微分

$$y_{\text{err}} = \frac{ax_{\max} + b - 1/x_{\max}}{1/x_{\max}}$$

且設定

$$\frac{dy_{\text{err}}}{dx} = 0$$

$$x_{\max} = 0.5 (x_2 + x_1)$$

五、發明說明 (11)

來求得。如可自圖3看出，雖然最壞情形之誤差出現於 $x_{\max}$ ，在二端點之誤差是零。誤差可散佈至該間段，且藉由朝著曲線A-B之點 $x_{\max}$ 向下"滑動"線C-D至與點 $x_{\max}$ 之距離變成原來之1/2可降低最大誤差成為原來值之大約1/2。此使得該線處於圖3所示之線E-F之位置。線E-F之斜率維持在a。線E-F之偏移b'可如下求得：

$$b' = b - 0.5 (ax_{\max} + b - 1/x_{\max})$$

本發明之方法提供一以常數a及b'為基礎之線性近似。檢視函數 $1/x$ 及 $ax+b$ 可發現斜率a將永遠是負的。為方便之故，在隨後之推導中a被當做是正的，且接著自b'減去 ax 。

針對介於值1及2間之不同間段的常數a及b'可儲存於一或更多之檢查表。特殊之常數對是藉由引數x之最高有效假數位元來加以選擇。間段之大小是允許誤差標準得以滿足之2的最大次方。對於小於 $1/2^{12}$ 之相對誤差而言，在範圍 $1 \leq x < 1.78125$ 之中，間段大小是1/32，而在範圍 $1.78125 \leq x < 2$ 之中，間段大小是1/16。接著藉由忽略隱含位元及使用假數之最高j有效位元以自儲存表選擇適當之常數a及b'來選擇一給定表之適當輸入項。如果準確度標準是相對誤差低於 $1/2^{12}$ ，則j=5。接著，估計之倒數y是以 $ax+b'$ 來加以計算。

藉由考慮精確度需求，截位，及捨入誤差，計算 $y = b' - ax$ 意謂一15-位元之相乘及相加。但是，該乘法可藉由認

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明 (12)

知，就所選之常數對而言， x 之最先五位元為已知，來降低至10位元。就一特定硬體實例而言，引數 x 可分成高階部份 x_H 及低階部份 x_L 。 x_H 部份包含用以定址該表之 j 位元。當產生該表時， a 及 x_H 之乘積可預先加以計算，自 b' 減去，及受到儲存。因此，就每一 x 間段而言，方程式 $y=b'-ax$ 可改寫為：

$$\begin{aligned} y &= b' - a(x_H + x_L) \\ &= (b - ax_H) - ax_L \\ &= b'' - ax_L \end{aligned}$$

請參看圖3， b'' 是 y_1 在線E-F之值。用於與 x_L 相乘之常數 a 可較 b'' 少 j 位元，因為 x_L 在大小上較 b'' 小 2^j 倍。在一較佳實例中，儲存於該等表之實際常數是 a 及 b'' 之捨入版本。當考慮實際之捨入及截位誤差時，常數 b'' 需要15位元以提供12位元之準確度。因為捨入 x_L 會增加執行時間， x_L 受到截位。在一示範建構中，斜率 a 乘以 x_L ，其中 x_L 包含緊接在 x 用以存取該表之五小數位元以後之最高9有效位元。該等常數是以十六進位之型態展示於表1。

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

錄

五、發明說明 (13)

表 1

間段	左邊緣	右邊緣	斜率	修正偏移
0	1.00000	1.03125	000003E1	00007FFC
1	1.03125	1.06250	000003A7	00007C1C
2	1.06250	1.09375	00000371	00007875
3	1.09375	1.12500	00000340	00007504
4	1.12500	1.15625	00000313	000071C4
5	1.15625	1.18750	000002EA	00006EB1
6	1.18750	1.21875	000002C4	00006BC8
7	1.21875	1.25000	000002A0	00006904
8	1.25000	1.28125	0000027F	00006664
9	1.28125	1.31250	00000261	000063E5
10	1.31250	1.34375	00000245	00006184
11	1.34375	1.37500	0000022A	00005F40
12	1.37500	1.40625	00000212	00005D16
13	1.40625	1.43750	000001FB	00005B04
14	1.43750	1.46875	000001E5	0000590A
15	1.46875	1.50000	000001D1	00005725
16	1.50000	1.53125	000001BE	00005554
17	1.53125	1.56250	000001AC	00005396
18	1.56250	1.59373	0000019B	000051EB
19	1.59375	1.62500	0000018B	0000504F
20	1.62500	1.65625	0000017C	00004EC4
21	1.65625	1.68750	0000016E	00004D48
22	1.68750	1.71875	0000015B	00004BD9
23	1.71875	1.75000	0000015B	00004A78
24	1.75000	1.78125	00000143	00004924
25	1.78125	1.81250	00000143	000047DB
26	1.81250	1.84375	0000012D	0000469C
27	1.84375	1.87500	0000012D	0000456F
28	1.87500	1.90625	0000011A	00004442
29	1.90625	1.93750	0000011A	00004328
30	1.93750	1.96875	00000108	0000420E
31	1.96875	2.00000	00000108	00004106

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

冰

五、發明說明 (14)

舉例而言，對於倍準度IEEE浮點數字，令x之小數位元編號成為x(1:52)，其中位元1是最高有效位元，且正好位於隱含位元1之右邊。位元x(1:5)是用以定址該表，且 x_L 等於x(6:14)，而x(6:14)是用以與自該表所讀取之值a相乘。如果乘法器使用一修正型布斯重複編碼器來減少進位-儲存-加法器之位準數目，則布斯重複編碼可與表存取平行進行，因而降低乘法器之有效延遲。乘法器產生一19-位元之乘積，但是只有自 b'' 減去最高10有效位元。該等位元校準於 b'' 之最低10位元。就前述範例而言，如果 $p(1:19)$ 是a及 x_L 之19-位元中間乘積，則使用下列減去來求得y：

$$y = 0.b''(1:15) - 0.00000p_1p_2p_3p_4p_5p_6p_7p_8p_9p_{10}$$

適當之邏輯電路可配備於處理器以建構前述功能，而熟悉本技術領域者應可明瞭該等邏輯電路。若要驗證此演算法，則可撰寫程式以測試假數之所有可能值，且所有值之結果皆落在所要之準確度以內。在表1中，開始於間段26之斜率行之該等常數受到複製。此種簡化是能夠使用1/16之間段於 $1.78125 \leq x < 2$ ，且同時符合準確度要求，之結果。

用於倒數估計所需之準確度決定表輸入項之數目及儲存值之位元數目。本文所用之準確度之定義類似於PowerPC™架構所用之定義。12位元之準確度是定義成為低於 $1/2^{12} = 1/4096$ 之相對誤差，亦即，

$$ABS\left(\frac{\text{estimate} - 1/x}{1/x}\right) \leq 1/4096$$

五、發明說明 (15)

本文所述之演算法符合此要求且，除此之外，通常更為有用，因為其可做為快速除法演算法之基礎。此方法降低乘法器之大小及延遲。當與一牛頓-瑞佛森疊代結合時，此演算法之12-位元準確度版本可藉由額外通過相乘-相加管線二次來提供一幾乎單準確度之IEEE相容型倒數。如果需要倍準確度之倒數，則以此演算法之直接延伸來提供14-位元準確度將是有利的。倍準確度之倒數接著需要二牛頓-瑞佛森疊代，或通過相乘-相加管線四次。

一般而言此演算法可導致計算一數字之倒數所需之時間獲得大幅改善，而只導致成本之相當微小增加。如在相關技術說明所述，若要利用傳統之檢查表技術來產生12正確之位元，則需要一包含2048輸入項之表。本文所述之技術無需該2048輸入項之表，而只需要一15-位元加法器，一10-位元乘以9-位元之乘法，及一需要少於32獨特輸入項對之表。在該等表輸入項對中，該等數字之一數字需要最多15位元，另一數字需要最多10位元，且以上二數字皆是定點數字。因此，此演算法大幅減少硬體，但可提供相當之效能。

圖4顯示一種用以建構本發明之方法。一全準確度輸入 x 區分成為高階部份 x_H 及低階部份 x_L 。該高階部份饋至檢查表200，而檢查表200提供第一常數(斜率)至布斯多工器202，且也提供第二常數(偏移)至乘法陣列及加法器204。該低階部份是做為布斯重複編碼器206之輸入，布斯重複編碼器206充當布斯多工器202之多工選擇。

五、發明說明 (16)

圖5顯示本發明之另一建構，且該建構運用一傳統浮點單元之乘法器之一些電路，而該傳統浮點單元包含頂乘法陣列208，多工器210，與底乘法陣列及加法器212。輸入值之高階部份再一次饋至檢查表214。表214再一次提供第一常數(斜率)至布斯多工器216，但是，在此建構中，第二常數(偏移)饋至多工器210。輸入值之低階部份仍然饋至布斯重複編碼器218，而布斯重複編碼器218充當布斯多工器216之多工選擇。

根據本發明之第二建構，且再一次參看圖3，該圖形現在可視為平方根倒數函數 $y=1/\sqrt{x}$ 之一區段。仍然假設 y 及 x 是經正規化之數字，且現在 $0.5 \leq x < 2$ 。如前所述，如何處理浮點指數為眾所知，所以在不喪失一般性之下本文只討論假數之產生。在圖3中，曲線A-B表示函數 $y=1/\sqrt{x}$ 之一區段。在點 x_1 及 x_2 之間，線C-D近似於 $1/\sqrt{x}$ 。線C-D之斜率 a 可由下列方程式求得：

$$a=(y_2-y_1)/(x_2-x_1)=(1/\sqrt{x_2}-1/\sqrt{x_1})/(x_2-x_1)。$$

偏移 b 則可由下列方程式來求得：

$$b=y_1-ax_1$$

發生於 $x_{\max}$ 之最大相對誤差 y_{err} 可藉由下列微分

$$y_{\text{err}} = \frac{ax_{\max} + b - 1/\sqrt{x_{\max}}}{1/\sqrt{x_{\max}}}$$

且設定

五、發明說明 (17)

$$\frac{dy_{err}}{dx} = 0$$

$$x_{max} = -b/(3a)$$

來求得。再一次，雖然最壞情形之誤差出現於 x_{max} ，在二端點之誤差是零，且誤差可散佈至該間段，且藉由朝著曲線 A-B 之點 x_{max} 向下"滑動"線 C-D 至與點 x_{max} 之距離變成原來之 1/2 可降低最大誤差成為原來值之大約 1/2。此使得該線處於圖 3 之線 E-F 所示之位置。線 E-F 之斜率維持在 a 。線 E-F 之偏移 b' 可如下求得：

$$b' = b - 0.5 (ax_{max} + b - 1/\sqrt{x_{max}})$$

本發明之方法再一次提供一以常數 a 及 b' 為基礎之線性近似。檢視函數 $1/\sqrt{x}$ 及 $ax+b$ 同樣可發現斜率 a 將永遠是負的。為方便之故，在隨後之推導中 a 被當做是正的，且接著自 b' 減去 ax 。

針對介於 0.5 及 2 間之不同間段的常數 a 及 b' 可儲存於許多檢查表。特定之常數對是藉由引數 x 之最高有效假數位元來加以選擇。間段之大小是允許誤差標準得以滿足之 2 的最大次方。對於小於 $1/2^{12}$ 之相對誤差而言，在範圍 $0.5 \leq x < 1$ 之中，間段大小是 $1/32$ ，而在範圍 $1 \leq x < 2$ 之中，間段大小是 $1/16$ 。間段大小之變化使得使用二表較為便利，且每一表皆具有 16 對之輸入項。第一表是用於引數 $0.5 \leq x < 1$ ，而第二表是用於引數 $1.0 \leq x < 2$ 。平方根倒數是以指數之最低有效位元為基礎藉由選擇適當來加以計算。接著藉由忽

五、發明說明 (18)

略隱含位元及使用假數之最高j有效位元以自儲存表選擇適當之常數a及b'來選擇一給定表之適當輸入項。如果準確度標準是相對誤差低於 $1/2^{12}$ ，則j=4。接著，估計之平方根倒數y是以 $ax+b'$ 來加以計算。

藉由考慮精確度需求，截位，及捨入誤差，計算 $y=b'-ax$ 意謂一15-位元之相乘及相加。但是，該相乘可再一次藉由認知，就所選之常數對而言，x之最先四位元及適當表為已知，來降低至10位元。就一特定硬體實例而言，引數x同樣可分成高階部份 x_H 及低階部份 x_L 。 x_H 部份包含表選擇位元及用以定址該表之j位元。當產生該表時，a及 x_H 之乘積可受到計算，自b'減去，及受到儲存。因此，就每一x間段而言，方程式 $y=b'-ax$ 可改寫為：

$$\begin{aligned} y &= b' - a(x_H + x_L) \\ &= (b - ax_H) - ax_L \\ &= b'' - ax_L \end{aligned}$$

請參看圖3， b'' 是 y_1 在線E-F之值。用於與 x_L 相乘之常數a可較 b'' 少j位元，因為 x_L 在大小上較 b'' 小 2^j 倍。在一較佳實例中，儲存於該等表之實際常數再一次是a及 b'' 之捨入版本。當考慮實際之捨入及截位誤差時，常數 b'' 需要16位元以提供12位元之準確度。因為捨入 x_L 會增加執行時間， x_L 受到截位。在一示範建構中，斜率a乘以 x_L ，其中 x_L 包含緊接在x用以存取該表之四小數位元以後之最高10有效位元。下文之表2是用於引數 $0.5 \leq x < 1$ 。表3是用於引

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明 (19)

數 $1.0 \leq x < 2$ 。該等常數是以十六進位之型態來受到展示。

表 2

間段	左邊緣	右邊緣	斜率(a)	修正偏移(b")
0	.50000	.53125	00000568	0000B4FD
1	.53125	.56250	000004F3	0000AF97
2	.56250	.59357	0000048D	0000AAA5
3	.59357	.62500	00000435	0000A1E4
4	.63500	.65625	000003E7	0000A1E4
5	.65625	.68750	000003A2	00009DFE
6	.68750	.71875	00000365	00009A5C
7	.71875	.75000	0000032E	000096F8
8	.75000	.78125	000002FC	000093CA
9	.78125	.81250	000002D0	000090CE
10	.81250	.84375	000002A8	00008DFE
11	.84375	.87500	00000283	00008B57
12	.87500	.90625	00000261	000088D4
13	.90625	.93750	00000243	00008673
14	.93750	.96875	00000226	00008431
15	.96875	1.00000	0000020B	0000820B

要建構於 $0.5 \leq x < 1.0$ 之修正斜率及偏移值

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明 (20)

表 3

間段	左邊緣	右邊緣	斜率	修正偏移
0	1.00000	1.06250	000003D2	00007FFA
1	1.06250	1.12500	00000380	00007C29
2	1.12500	1.18750	00000338	000078AA
3	1.18750	1.25000	000002F9	00007572
4	1.25000	1.31250	000002C2	00007279
5	1.31250	1.37500	00000292	00006FB7
6	1.37500	1.43750	00000266	00006D26
7	1.43750	1.50000	0000023F	00006AC0
8	1.50000	1.56250	0000021D	00006881
9	1.56250	1.62500	000001FD	00006665
10	1.62500	1.68750	000001E1	00006468
11	1.68750	1.75000	000001C7	00006287
12	1.75000	1.81250	000001AF	000060C1
13	1.81250	1.87500	00000199	00005F12
14	1.87500	1.93750	00000185	00005D79
15	1.93750	2.00000	00000173	00005BF4

要建構於 $1.0 \leq x < 2.0$ 之修正斜率及偏移值

舉例而言，對於倍準度之IEEE浮點數字，再一次令 x 之小數位元編號成為 $x(1:52)$ ，其中位元1是最高有效位元，且正好位於隱含位元1之右邊。指數之最低有效位元是用以選擇該表，位元 $x(1:4)$ 是用以定址該表，且 x_L 等於 $x(5:14)$ ，而 $x(5:14)$ 是用以與自該表所讀取之值 a 相乘。如果乘法器使用一修正型布斯重複編碼器來減少進位-儲存-加法器之位準數目，則布斯重複編碼同樣可與表存取平行進行，因而降低乘法器之有效延遲。乘法器產生一20-位元之乘積，但是只有自 b'' 減去最高11有效位元。該等位元校準於 b'' 之最低11位元。就前述範例而言，如果 $p(1:20)$

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

五、發明說明(21)

是 a 及 x_L 之 20-位元中間乘積，則使用下列減法來求得 y ：

$$y = b''(0).b''(1:15) - 0.0000p_1p_2p_3p_4p_5p_6p_7p_8p_9p_{10}p_{11}$$

其中 $b''(0)=1$ ，若 $0.5 \leq x < 1$ ，且 $b''(0)=0$ ，若 $1.0 \leq x < 2$ 。適當之邏輯電路可再一次配備於處理器以建構前述功能，而熟悉本技術領域者應可明瞭該等邏輯電路。若要驗證此演算法，則可撰寫程式以測試假數之所有可能值，且所有值之結果皆要落在所要之準確度以內。

展現於本文之平方根倒數之演算法再一次符合 12-位元之準確度，且同時降低乘法器之大小及延遲。當與一牛頓-瑞佛森疊代結合時，此演算法可藉由額外通過相乘-相加管線三次來提供一幾乎全單準度之 IEEE 相容型平方根。如果需要倍準度之平方根倒數，則以此演算法之直接延伸來提供 14-位元準確度將是有利的。倍準度之平方根倒數接著需要二牛頓-瑞佛森疊代，或通過相乘-相加管線六次。

一般而言此演算法再一次可導致計算一數字之平方根倒數所需之時間獲得大幅改善，而只導致成本之相當微小增加。本文所述之技術無需該 2048 輸入項之表，而只需一 16-位元加法器，一 11-位元乘以 10-位元之乘法，及一需要 32 獨特輸入項對之表。在該等表輸入項對中，該等數字之一數字需要最多 16 位元，另一數字需要最多 11 位元，且以上二數字皆是定點數字。因此，此演算法同樣大幅減少硬體，但可提供相當之效能。圖 4 及 5 可利用前述方式之類似方式來建構本發明以應用於平方根倒數函數。

五、發明說明 (22)

可利用許多不同之方式來提供平方根函數。一種方式，如前所述，是提供一引數之平方根倒數，且接著使得該值乘以該引數以求得其之平方根。另一方式是使用直接提供平方根之表。剛才所述用以提供平方根倒數之方法，藉由使用二表，也應可藉由使用該二表之適當值來提供平方根函數，但要加上一微小修改。就平方根函數而言，輸入之假數之低階部份及第一常數之乘積必須加至第二常數，而非自第二常數減去。另外，存在其他基本函數，例如一數字之平方及一些超越函數，且該等基本函數可透過本發明之協助來加以提供。

相同系統也可能同時需要倒數及平方根倒數函數。不一定要提供分別之硬體給該二函數。該表可擴充成為包含使用本發明所需之所有函數，以致圖4及5所示之其他硬體，例如布斯重複編碼器，多工器，乘法陣列，及加法器，皆可為各種函數所共享。

雖然一直參照特定實例來說明本發明，此種說明不應視為一種限制。一旦閱讀本發明之說明，熟悉本技術領域者應可明瞭所述實例之各種修改，以及本發明之其他實例。例如，雖然本發明特別適用於大型電腦系統，例如本文描述之電腦系統，本發明同樣適用於較小，有限用途之電腦系統，例如掌上型計算器。因此構思在不脫離本發明之精神或範疇之下可進行該等修改，且本發明之精神或範疇定義於附加之申請專利範圍。

四、中文發明摘要 (發明之名稱： 基本函數之高準確度預估)

本發明提出一種利用電腦系統來估計一輸入值之平方根，平方根倒數，與倒數的改良方法。該輸入值，在正規化之後，是用以自一表選擇一對常數。該二常數是以該輸入值之每一間段之該函數的一線性近似為基礎，且受到偏移以降低一給定間段之最大誤差。估計之函數是藉由相加或相減經正規化之輸入值之一部份及第一常數的乘積與第二常數來加以計算。在一建構中，該輸入值是在範圍 $1 \leq x < 2$ 之中受到正規化，且使用一檢查表，且該檢查表之間段大小為 $1/32$ 。在另一較佳實例中，只有假數之低階部份是用於相乘-相加運算，以減少所需之位元數目(假數之高階部份是用以自該表選擇該等常數)。在另一建構中，該輸入值是在範圍 $0.5 \leq x < 2$ 之中受到正規化，且使用二檢查表，而

英文發明摘要 (發明之名稱：

"HIGH ACCURACY ESTIMATES OF
ELEMENTARY FUNCTIONS"

An improved method of estimating the square root, reciprocal square root, and reciprocal of an input value in a computer system. The input value, after being normalized, is used to select a pair of constants from a table. The constants are based on a linear approximation of the function for each interval of the input value, offset to reduce a maximum error value for a given interval. The estimated function is calculated by adding or subtracting the product of a part of the normalized input value and the first constant from the second constant. In one implementation, the input value is normalized within the range $1 \leq x < 2$, and one lookup table is used, having an

四、中文發明摘要(發明之名稱:)

第一檢查表在範圍 $0.5 \leq x < 1$ 之中其之間段大小為 $1/32$ ，且第二檢查表在範圍 $1 \leq x < 2$ 之中其之間段大小為 $1/16$ 。

(請先閱讀背面之注意事項再填寫本頁各欄)

裝

訂

線

英文發明摘要(發明之名稱:)

interval size of $1/32$. In a further preferred embodiment, only a lower order part of the mantissa is used in the multiply-add operation, to reduce the number of bits required (the high order part of the mantissa is used to select the constants from the table). In another implementation, the input value is normalized within the range $0.5 \leq x < 2$, and two lookup tables are used, a first table having an interval size of $1/32$ for the range $0.5 \leq x < 1$, and a second table having an interval size of $1/16$ for the range $1 \leq x < 2$.

六、申請專利範圍

1. 一種利用電腦系統來估計一輸入值之倒數的方法，該方法包含下列步驟：

正規化該輸入值以產生一經正規化之輸入值；

自具有多個間段之一表選擇第一常數及第二常數，以經正規化之輸入值為基礎，其中該等第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/x$ 的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值；及

利用處理器來計算一估計之倒數，而此是藉由結合經正規化之輸入值及第一常數之乘積與第二常數來達成。

2. 如申請專利範圍第1項之方法，其中該正規化步驟正規化在範圍 $1 \leq x < 2$ 中之輸入值。

3. 如申請專利範圍第1項之方法，其中：

經正規化之輸入值具有一假數；且

該選擇步驟以該假數之最高有效位元為基礎來選擇第一及第二常數。

4. 如申請專利範圍第1項之方法，其中第一及第二常數具有足夠之準確度以致估計之倒數具有低於 $1/4096$ 之相對誤差。

5. 如申請專利範圍第1項之方法，其中處理器具有一乘法器，且該乘法器使用一修正型布斯重複編碼器來減少進位-儲存-加法器之位準數目，且進一步包含下列步驟：平行使用布斯重複編碼及該表以降低乘法器之有效延遲。

6. 如申請專利範圍第1項之方法，其中該表之至少二間段

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

六、申請專利範圍

對於第一常數具有相同值。

7. 如申請專利範圍第3項之方法，其中第二常數進一步是以一給定間段之左及右邊緣之值為基礎，且該選擇步驟是藉由使用該假數之高階部份以降低相乘-相加運算所需之位元數目來達成。
8. 一種用於電腦系統之處理器，該種處理器包含：
多個暫存器；
連接至該等暫存器之多個邏輯單元，而該等邏輯單元包括一允許相乘-相加運算之浮點執行單元；
一用以估計一經正規化之輸入值之倒數的檢查表，且該輸入值位於該等暫存器之一，且該檢查表具有多個間段及相關之第一及第二常數對，且該等第一及第二常數是用以輸入至該浮點執行單元，且該等第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/x$ 的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值，以致可藉由結合經正規化之輸入值及第一常數之乘積與第二常數來估計該倒數。
9. 如申請專利範圍第8項之處理器，且該處理器進一步包含裝置以正規化在該檢查表之一範圍以內之一輸入值來產生經正規化之輸入值。
10. 如申請專利範圍第8項之處理器，其中該檢查表提供該等第一及第二常數，且該等第一及第二常數具有足夠之準確度以致估計之倒數具有低於 $1/4096$ 之相對誤差。
11. 如申請專利範圍第8項之處理器，其中經正規化之輸入

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

評

六、申請專利範圍

值具有一假數部份，且該第二常數進一步是以一給定間段之左及右邊緣之值為基礎，以致經正規化之輸入值及第一常數之乘積可藉由使用該假數之低階部份來加以計算，似降低相乘-相加運算所需之位元數目。

12. 如申請專利範圍第8項之處理器，其中該表之至少二間段對於第一常數具有相同值。

13. 一種利用一電腦系統來估計一輸入值之平方根倒數的方法，且該電腦系統具有一處理器，該方法包含下列步驟：

正規化該輸入值以產生一經正規化之輸入值；

自具有多個間段之一表選擇第一常數及第二常數，以經正規化之輸入值為基礎，其中第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/\sqrt{x}$ 的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值；及

利用處理器來計算一估計之平方根倒數，且此是藉由結合經正規化之輸入值及第一常數之乘積與第二常數來達成。

14. 如申請專利範圍第13項之方法，其中該正規化步驟正規化在範圍 $0.5 \leq x < 2$ 中之輸入值。

15. 如申請專利範圍第13項之方法，其中：

經正規化之輸入值具有一假數；且

該選擇步驟以該假數之最高有效位元為基礎來選擇第一及第二常數。

16. 如申請專利範圍第13項之方法，其中該選擇步驟藉由首

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

以

六、申請專利範圍

先自多個不同表選擇該表來選擇第一及第二常數，且該等多個不同表對應於經正規化之輸入值之不同間段。

17. 如申請專利範圍第13項之方法，其中第一及第二常數具有足夠之準確度以致估計之平方根倒數具有低於 $1/4096$ 之相對誤差。

18. 如申請專利範圍第13項之方法，且該方法進一步包含下列步驟：

藉由相乘輸入值及估計之平方根倒數來計算輸入值之估計平方根。

19. 如申請專利範圍第13項之方法，其中處理器具有一乘法器，且該乘法器使用一修正型布斯重複編碼器來減少進位-儲存-加法器之位準數目，且進一步包含下列步驟：平行使用布斯重複編碼及該表以降低乘法器之有效延遲。

20. 如申請專利範圍第15項之方法，其中第二常數進一步是以一給定間段之左及右邊緣之值為基礎，且該選擇步驟是藉由使用該假數之高階部份來達成，以降低相乘-相加運算所需之位元數目。

21. 如申請專利範圍第16項之方法，其中：

該正規化步驟正規化在範圍 $0.5 \leq x < 2$ 中之輸入值；

第一及第二常數具有足夠之準確度以致估計之平方根倒數具有低於 $1/4096$ 之相對誤差；及

該選擇步驟自第一表及第二表選擇第一常數及第二常數，且該第一表在範圍 $0.5 \leq x < 1$ 中其之間段大小是

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

六、申請專利範圍

1/32，而第二表在範圍 $1 \leq x < 2$ 中其之間段大小是 1/16。

22. 如申請專利範圍第 18 項之方法，且該方法進一步包含下列步驟：

藉由使用估計之平方根倒數做為一牛頓-瑞佛森疊代之第一近似來改善估計之平方根之準確度。

23. 如申請專利範圍第 20 項之方法，其中該選擇步驟藉由首先自多個不同表選擇該表來選擇第一及第二常數，且該等多個不同表對應於經正規化之輸入值之不同間段。

24. 一種用於電腦系統之處理器，且該種處理器包含：

多個暫存器；

連接至該等暫存器之多個邏輯單元，而該等邏輯單元包括一允許相乘-相加運算之浮點執行單元；

一用以估計一經正規化之輸入值之平方根倒數的檢查表，且該輸入值位於該等暫存器之一，且該檢查表具有多個間段及相關之第一及第二常數對，且該等第一及第二常數是用以輸入至該浮點執行單元，且該等第一及第二常數是以使用一給定間段之左及右邊緣之函數 $1/\sqrt{x}$ 的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值，以致可藉由結合經正規化之輸入值及第一常數之乘積與第二常數來估計該平方根倒數。

25. 如申請專利範圍第 24 項之處理器，且該處理器進一步包含裝置以正規化在該檢查表之一範圍以內之一輸入值以產生經正規化之輸入值。

26. 如申請專利範圍第 24 項之處理器，其中該檢查表是第一

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

錄

六、申請專利範圍

檢查表，且進一步包含第二檢查表，而該第二檢查表具有不同於該第一檢查表之間段。

27. 如申請專利範圍第24項之處理器，其中該檢查表提供該等第一及第二常數，且該等第一及第二常數具有足夠之準確度以致估計之平方根倒數具有低於 $1/4096$ 之相對誤差。
28. 如申請專利範圍第24項之處理器，其中經正規化之輸入值具有一假數部份，且該第二常數進一步是以一給定間段之左及右邊緣之值為基礎，以致經正規化之輸入值及第一常數之乘積可藉由使用該假數之低階部份來加以計算，似降低相乘-相加運算所需之位元數目。
29. 如申請專利範圍第24項之處理器，且該處理器進一步包含裝置以藉由相乘輸入值及估計之平方根倒數來計算經正規化之輸入值之平方根。
30. 如申請專利範圍第26項之處理器，其中：
- 該正規化輸入值位於範圍 $0.5 \leq x < 2$ 以內；
- 第一及第二常數具有足夠之準確度以致估計之平方根倒數具有低於 $1/4096$ 之相對誤差；
- 第一表是用於範圍 $0.5 \leq x < 2$ 且其之間段大小是 $1/32$ ；且
- 第二表是用於範圍 $1 \leq x < 2$ 且其之間段大小是 $1/16$ 。
31. 如申請專利範圍第29項之處理器，且該處理器進一步包含裝置以藉由使用估計之平方根倒數做為一牛頓-瑞佛森疊代之第一近似來改善估計之平方根之準確度。
32. 一種利用電腦系統來估計一具有非線性部份之基本函數

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

冰

六、申請專利範圍

的方法，且該電腦系統具有一處理器，且該方法包含下列步驟：

正規化一輸入值以產生一經正規化之輸入值；

自具有多個間段之一表選擇第一常數及第二常數，以經正規化之輸入值為基礎，其中第一及第二常數是以使用一給定間段之左及右邊緣之基本函數非線性部份的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值；及

在處理器之相乘-相加運算中藉由使用經正規化之輸入值，第一常數，及第二常數來計算一估計之輸出值，其中該選擇步驟進一步降低相乘-相加運算所需之位元數目。

33. 一種用於電腦系統之處理器，且該種處理器包含：

多個暫存器；

連接至該等暫存器之多個邏輯單元，而該等邏輯單元包括一允許相乘-相加運算之浮點執行單元；

一以位於該等暫存器之一暫存器之經正規化輸入值為基礎來提供具有非線性部份之基本函數之估計的檢查表，且該檢查表具有多個間段及相關之第一及第二常數對，且該等第一及第二常數是用以輸入至該浮點執行單元，且該等第一及第二常數是以使用一給定間段之左及右邊緣之基本函數非線性部份的線性近似為基礎，且受到偏移以降低該給定間段之最大誤差值，且受到選擇以降低一相乘-相加運算所需之位元數目，且該相乘-相加運算計算一估計之輸出值。

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

外

434507

87109037

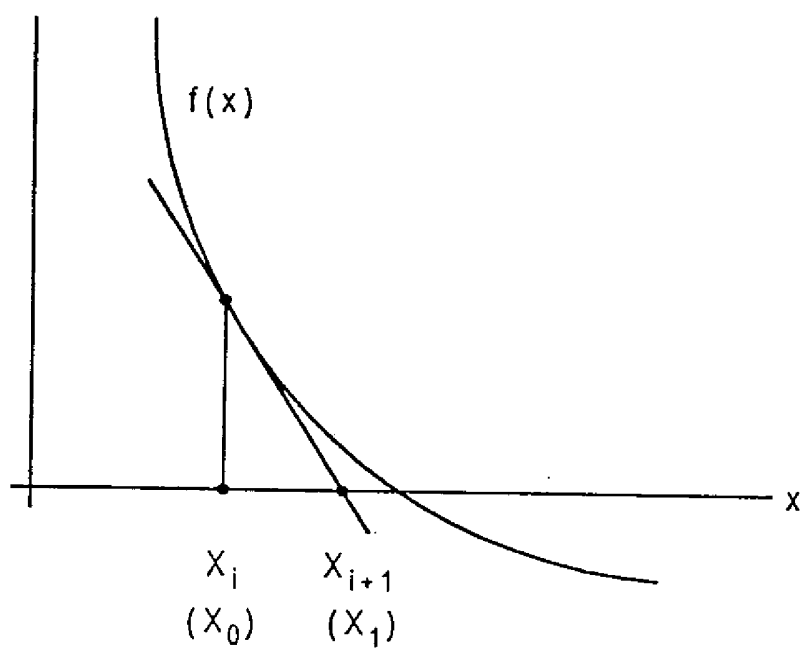


圖 1

以前技術

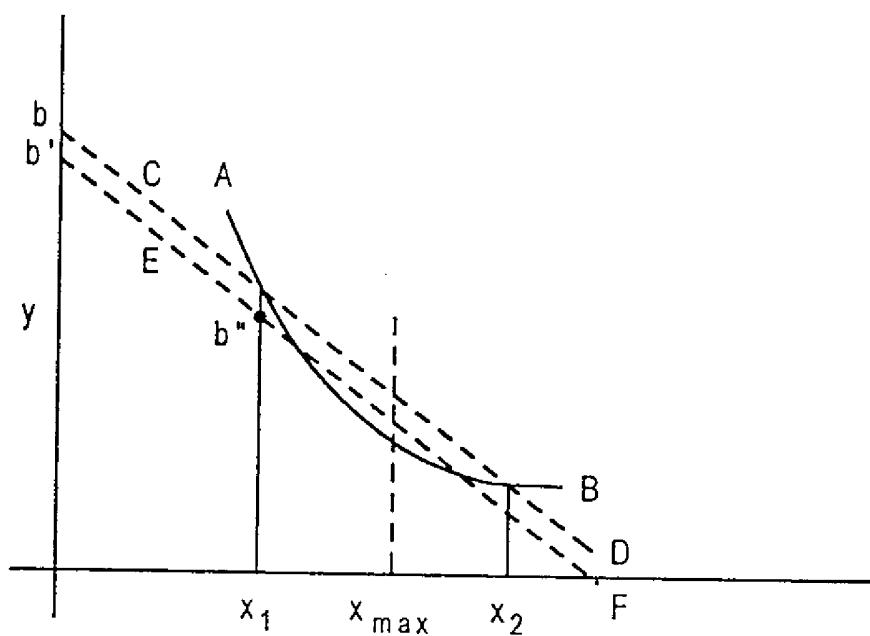


圖 3

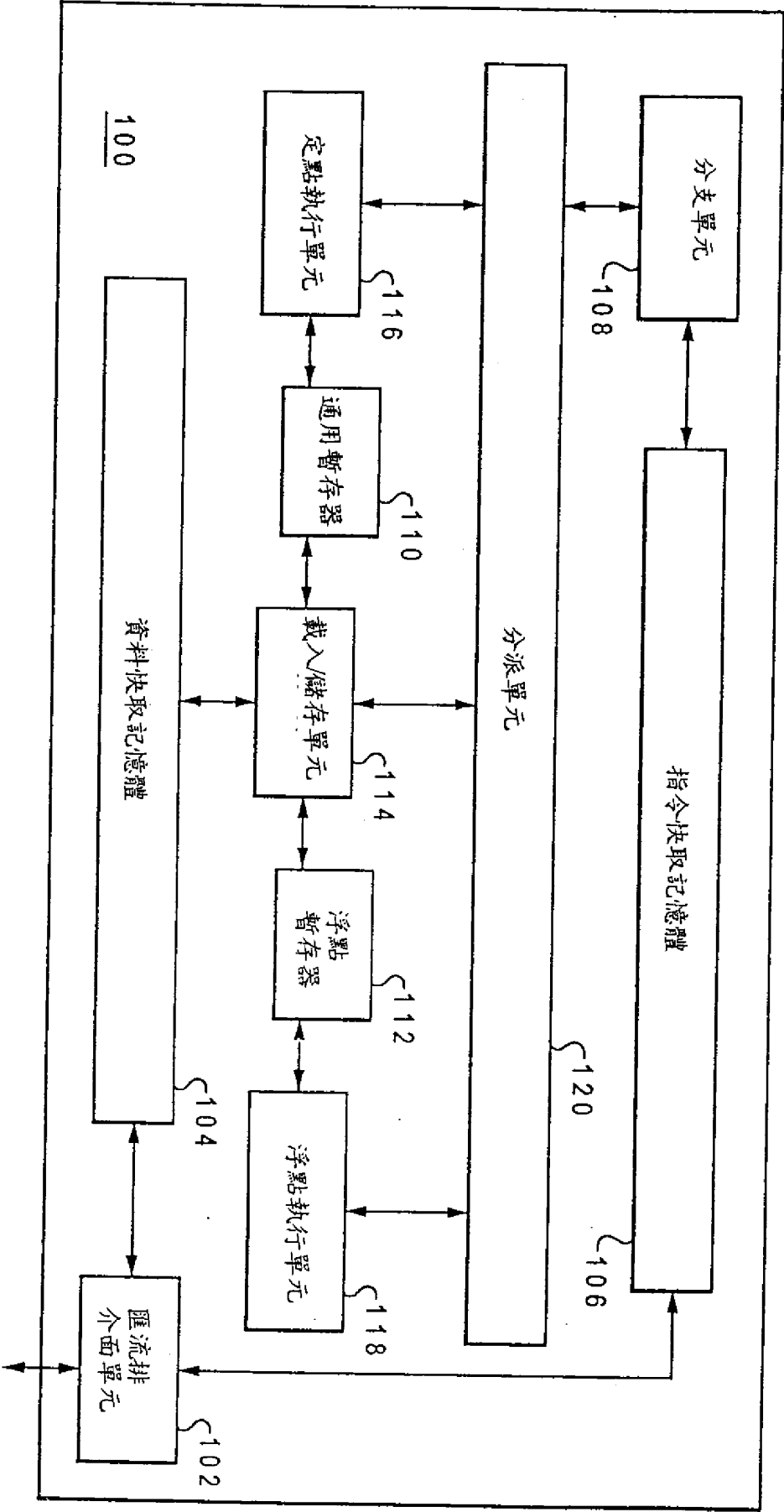


圖 2

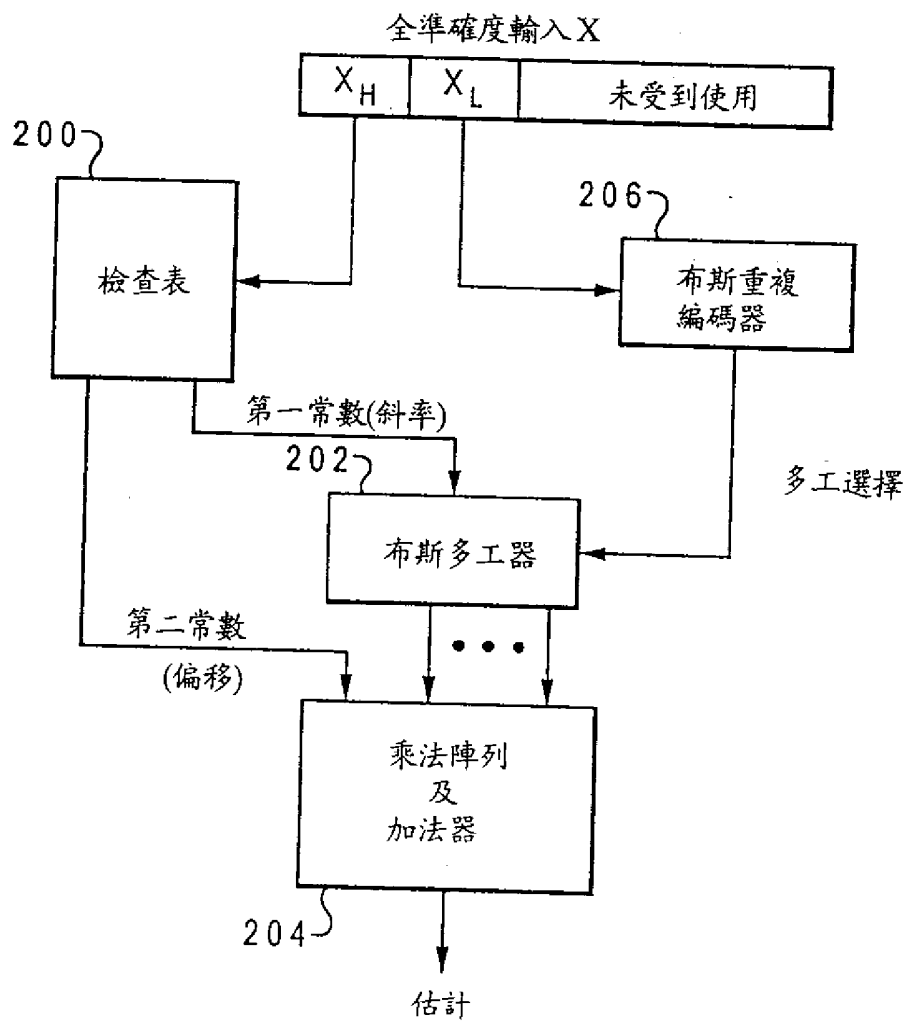


圖 4

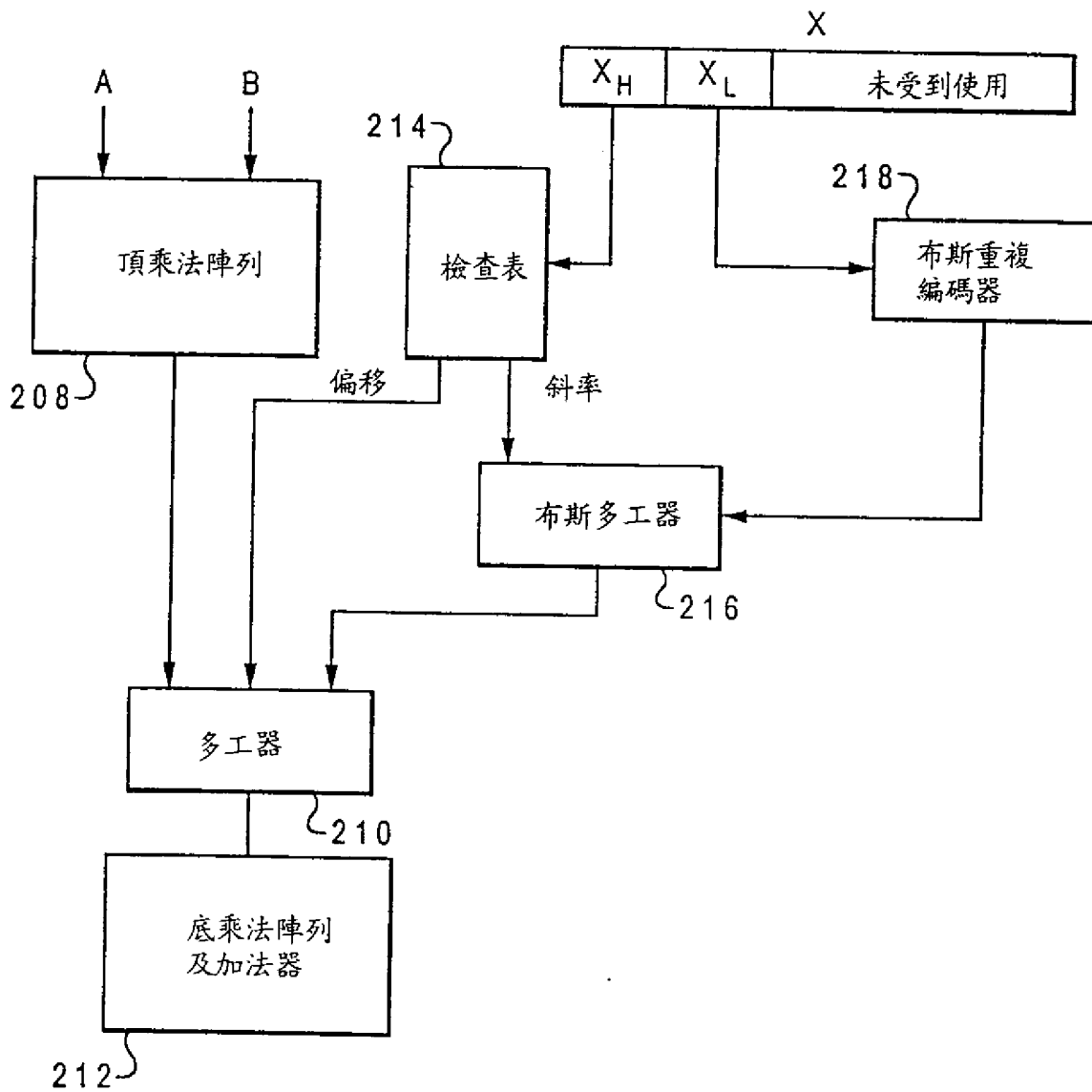


圖 5