202407538

(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 202407538 A

(43)公開日： 中華民國 113 (2024) 年 02 月 16 日

(21)申請案號：112126839 (22)申請日： 中華民國 112 (2023) 年 07 月 19 日

(51)Int. Cl.： *G06F9/30 (2018.01)* *G06F9/06 (2006.01)*

(30)優先權：2022/08/05 印度 202211044855

(71)申請人：英商ＡＲＭ股份有限公司 (英國) ARM LIMITED (GB)
英國

(72)發明人：霍利 約翰麥克 HORLEY, JOHN MICHAEL (GB)；拉特蘭 馬克賽爾林 RUTLAND, MARK SALLING (GB)；克拉斯克 西蒙約翰 CRASKE, SIMON JOHN (GB)；范吉雷迪 馬杜蘇達那雷迪 VANGIREDDY, MADHUSUDANA REDDY (IN)

(74)代理人：李世章；彭國洋

申請實體審查：無 申請專利範圍項數：23 項 圖式數：14 共 63 頁

(54)名稱

無操作相容指令

(57)摘要

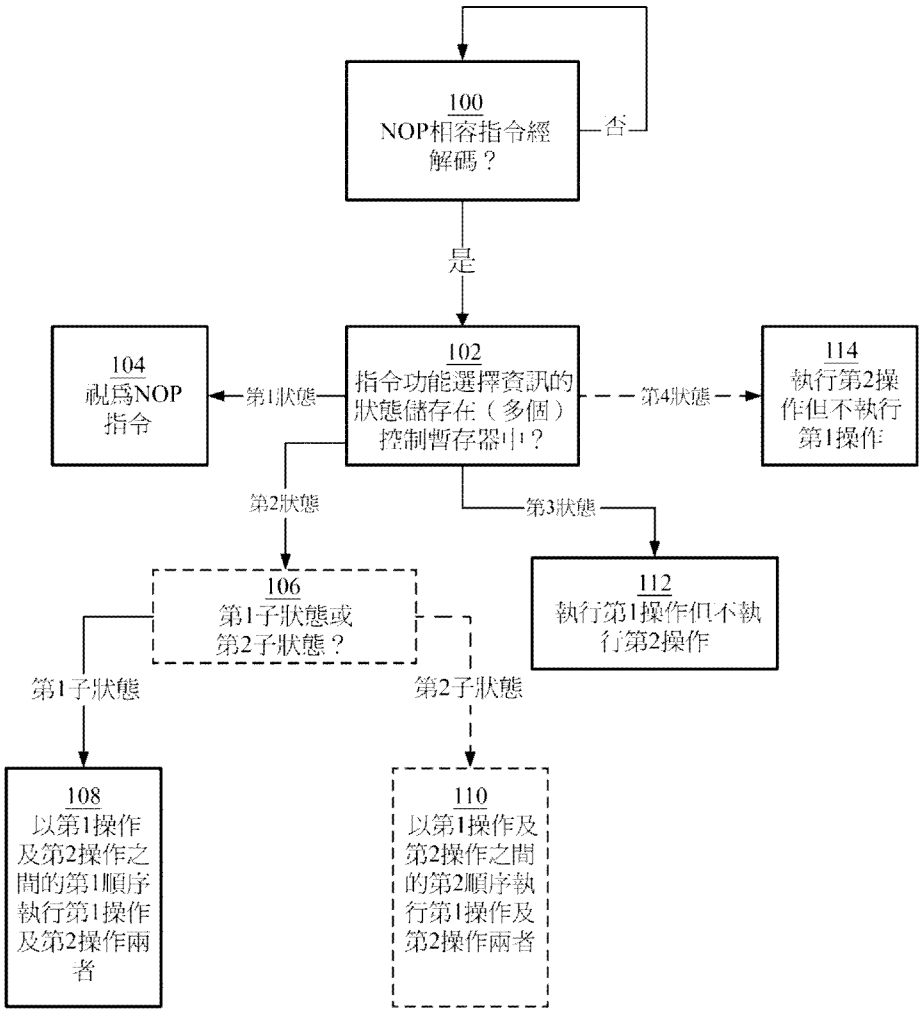
一種設備包含：一指令解碼器，以解碼指令；處理電路系統，以回應於該等指令藉由該指令解碼器的解碼而執行資料處理；及至少一控制暫存器，以指定指令功能選擇資訊。回應於一無操作相容指令，該指令解碼器經組態以控制該處理電路系統以：當由該至少一控制暫存器指定之該指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

An apparatus comprises an instruction decoder to decode instructions; processing circuitry to perform data processing in response to decoding of the instructions by the instruction decoder; and at least one control register to specify instruction-function-selecting information. In response to a no-operation-compatible instruction, the instruction decoder is configured to control the processing circuitry to: treat the no-operation-compatible instruction as a no-operation instruction, when the instruction-function-selecting information specified by the at least one control register is in a first state; perform both a first operation and a second operation, when the instruction-function-selecting information specified by the at least one control register is in a second state; and perform the first operation but not the second operation, when the instruction-function-selecting information specified by the at least one control register is in a third state.

指定代表圖：

符號簡單說明：

100,102,104,106,108,110,112,114:步驟



【圖4】

202407958

【發明摘要】

【中文發明名稱】　無操作相容指令

【英文發明名稱】　NO-OPERATION-COMPATIBLE INSTRUCTION

【中文】

一種設備包含：一指令解碼器，以解碼指令；處理電路系統，以回應於該等指令藉由該指令解碼器的解碼而執行資料處理；及至少一控制暫存器，以指定指令功能選擇資訊。回應於一無操作相容指令，該指令解碼器經組態以控制該處理電路系統以：當由該至少一控制暫存器指定之該指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【英文】

An apparatus comprises an instruction decoder to decode instructions; processing circuitry to perform data processing in response to decoding of the instructions by the instruction decoder; and at least one control register to specify instruction-function-selecting information. In response to a no-operation-compatible instruction, the instruction decoder is configured to control the processing circuitry to: treat the no-operation-compatible instruction as a no-operation instruction, when the instruction-function-selecting information specified by the at least one control register is in a first state; perform both a first operation and a second

operation, when the instruction-function-selecting information specified by the at least one control register is in a second state; and perform the first operation but not the second operation, when the instruction-function-selecting information specified by the at least one control register is in a third state.

【指定代表圖】圖4

【代表圖之符號簡單說明】

100,102,104,106,108,110,112,114:步驟

【特徵化學式】無

【發明說明書】

【中文發明名稱】　無操作相容指令

【英文發明名稱】　NO-OPERATION-COMPATIBLE INSTRUCTION

【技術領域】

【0001】　本技術係關於資料處理領域。

【先前技術】

【0002】　一種資料處理設備具有處理電路系統，用以回應於指令解碼器解碼的指令而執行資料處理。指令編碼的格式及由各指令所表示的功能可根據指令集架構(instruction set architecture, ISA)定義。ISA代表製造用於給定處理器實施方案的處理硬體之硬體製造商與寫入程式碼以在該硬體上執行的軟體開發者之間的同意構架，使得根據ISA編寫的程式碼將在支援ISA之硬體上正確地運作。選擇由ISA及其編碼支援之指令可存在設計挑戰。當規劃ISA的指令定義時，由ISA設計者所做出的設計決策可具有在執行特定程式時由處理硬體達成之對於真實世界效能之顯著的影響。

【發明內容】

【0003】　至少一些實例提供一種設備，其包含：一指令解碼器，以解碼指令；處理電路系統，以回應於該等指令藉由該指令解碼器的解碼而執行資料處理；及至少一控制暫存器，以指定指令功能選擇資訊；其中：回應於一無操作相容指令，該指令解碼器經組態以控制該處理電路系統以：當由該至少一控制暫存器指定之該指令功能選擇資訊在一第一狀態中

時，將該無操作相容指令視為一無操作指令；當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【0004】　至少一些實例提供一種方法，其包含：解碼指令；及回應於該等指令之解碼而執行資料處理；其中：回應於一無操作相容指令之解碼，該方法包含：當由至少一控制暫存器指定之指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【0005】　至少一些實例提供一種包含指令的電腦程式，當該等指令由一主機資料處理設備執行時，控制該主機資料處理設備提供用於執行目標碼的一指令執行環境，該電腦程式包含：指令解碼程式邏輯，以解碼該目標碼之指令；及暫存器仿真程式邏輯，以將資料維持在該主機資料處理設備的儲存電路系統中，以仿真至少一控制暫存器用於指定指令功能選擇資訊；其中：回應於一無操作相容指令，該指令解碼程式邏輯經組態以控制該主機資料處理設備以：當該指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；當該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【0006】 該電腦程式可儲存在一儲存媒體上。該儲存媒體可係一非暫時性儲存媒體或一暫時性儲存媒體。

【圖式簡單說明】

【0007】 本技術的進一步態樣、特徵、及優點將由於結合附圖閱讀的以下實例描述而顯而易見，在該等附圖中：

〔圖 1〕示意地繪示資料處理設備的一實例；

〔圖 2〕繪示設備之暫存器的實例；

〔圖 3〕繪示無操作相容（NOP 相容(no-operation-compatible)）指令之實例；

〔圖 4〕繪示處理 NOP 相容指令之方法；

〔圖 5〕繪示功能呼叫及功能返回的實例；

〔圖 6〕繪示保護控制堆疊(GCS)推送操作及 GCS 取出操作的實例；

〔圖 7〕繪示認證碼產生操作的實例；

〔圖 8〕繪示認證碼檢查操作的實例；

〔圖 9〕及〔圖 10〕繪示以不同順序執行 GCS 推送操作及認證碼產生操作的實例；

〔圖 11〕及〔圖 12〕繪示以不同順序執行 GCS 取出操作及認證碼檢查操作的實例；

〔圖 13〕繪示基於記憶體屬性資料檢查記憶體存取是否允許的步驟；及

〔圖 14〕繪示模擬實例。

【實施方式】

【0008】　一種設備包含：一指令解碼器，以解碼指令；處理電路系統，以回應於該等指令藉由該指令解碼器的解碼而執行資料處理；及至少一控制暫存器，以指定指令功能選擇資訊。回應於一無操作相容（NOP相容）指令，該指令解碼器控制該處理電路系統以：當由該至少一控制暫存器指定之該指令功能選擇資訊在一第一狀態中時，將該NOP相容指令視為一無操作(NOP)指令；當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【0009】　NOP相容指令可用於實施一些可選之操作，若執行，其可有用，但對於使用指令之軟體程式的正確運作並非關鍵。例如，第一操作及第二操作可係提供可選的安全增強或效能提示操作的操作，其可有助於改善安全及/或效能，即使對於獲得程式之正確運作結果並非必要。彼等增強可不必然需要，且因此所欲的是能夠在執行包括用於執行第一操作及第二操作中之一者或兩者的指令之一序列程式碼的給定實例上控制是否該等操作實際上被執行。未執行該等操作可幫助節省電力或允許分析相同程式碼將如何在不支援可選之操作的傳統硬體上運行。藉由基於儲存在至少一控制暫存器中的指令功能選擇資訊來控制指令的功能，相同程式碼可在不同使用情境中執行，其中取決於指令功能選擇資訊的目前值，第一操作及第二操作的不同結果（若有）係回應於指令由處理電路系統執行。

【0010】 當指令功能選擇資訊係在第一狀態時，該指令動作如NOP指令。NOP指令可係其執行除了程式計數器的變化以外不會導致架構狀態的任何變化之指令，程式計數器的變化可隱含在從NOP指令至NOP之後的下一指令依序前進之程式流程（不具有分支）中。因此，軟體開發者具有關閉由第一及第二操作一起代表之特徵的選項。此可有用，例如，其中第一操作及第二操作對應於在更新的ISA版本中引入的特徵，其可能無法在支援較舊ISA版本的傳統設備上可用。另一種使用情況可係第一操作及第二操作可不總是需要（例如，當針對較低安全使用情況執行程式碼時，在執行包括NOP相容指令的程式碼序列之前，可藉由設定指令功能選擇資訊至第一狀態，而關閉第一及/或第二操作代表之安全增強，以節省電力）。

【0011】 當指令功能選擇資訊係在第二狀態時，處理電路系統回應於NOP相容指令而執行第一操作及第二操作兩者。當指令功能選擇資訊係在第三狀態時，處理電路系統回應於NOP相容指令而執行第一操作，但不執行第二操作。通常，為了提供是否由程式碼序列執行第一操作及第二操作的不同組合，吾人將預期各操作編碼在單獨指令中，使得執行兩個操作之組合將需要兩個不同指令，而若需要此等操作的僅一者，則此等指令的僅一者將包括在軟體碼中。然而，利用上文所述之NOP相容指令，第一及第二操作兩者可回應相同指令而執行（以及具有相同指令被執行的選項係僅僅第一操作被執行，但不執行第二操作）。因此，用於包括NOP相容指令之程式碼之部分的完全相同程式碼二元可以不同方式動作，取決於執行程式碼的該部分之前指令功能選擇資訊已如何執行。相較於提供分別對應

於第一操作及第二操作之兩個不同的NOP相容指令的實施方案，此提供更有效的ISA編碼。首先，此避免ISA需要使用高達兩個不同的指令編碼，用於分別對應於第一操作及第二操作之不同的NOP相容指令，節省可用於另一種類型的指令的編碼，且因此改善使用該其他類型指令的程式碼的效能（相較於需要將該其他類型指令的操作分割成多個較簡單的指令來說）。再者，使用NOP相容指令的軟體程式碼將僅需要處理電路系統使用單一指令槽，用於處理管線的提取、解碼、發布、及執行階段中的指令（以及指令快取記憶體或用於儲存正在執行的程式碼的其他儲存器中的指令），其節省由實施不同操作的另一指令使用的指令槽，且因此可改善效能，以及改善快取記憶體中及記憶體中的程式碼儲存密度。

【0012】　因此，上文描述之NOP相容指令可幫助改善ISA編碼對於存在可能不總是需要的可選操作之使用情況之效能及效率。

【0013】　在一些實例中，NOP相容指令可不支援執行第二操作而不執行第一操作的選項。例如，指令功能選擇資訊的任何其餘編碼（除上文所描述之第一、第二及第三狀態之外）可用以控制第一或第二操作如何執行的其他特徵（例如，調整第一操作或第二操作的行為之控制參數），而非用以指示第二操作應被執行而第一操作則否。

【0014】　然而，在其他實例中，當該指令功能選擇資訊在一第四狀態中時，回應於該NOP相容指令，該指令解碼器可控制該處理電路系統，以執行該第二操作，但不執行該第一操作。此可尤其有用，使得NOP相容指令可用以執行該第一操作但不執行該第二操作，或者執行該第二操作但不執行該第一操作，或者兩操作一起執行，或可動作如NOP，使得第一操

作、第二操作皆不可執行。因此，可使用指令功能選擇資訊以獨立開啟及關閉第一操作及第二操作中之各者。此在若第一操作及第二操作中之各者經編碼為不同的NOP相容指令之情況中對於選擇哪個操作被執行提供相同的彈性，但具有更高效率的指令編碼。

【0015】 在一些實例中，當該指令功能選擇資訊係在該第二狀態中時，該處理電路系統可基於該指令功能選擇資訊控制施加該第一操作與該第二操作的一相對順序。例如，可在第二操作之前執行第一操作，或第二操作可在第一操作之前執行，或兩個操作可並行執行。指令功能選擇資訊可用以在二或更多個此等選項之間選擇。此可有用，因為第一操作與第二操作之間的一個順序可比另一個順序具有優點。例如，第一順序可提供更大的安全性，但第二順序可對效能更有效率。因此，提供控制狀態，其允許所使用之相對順序之組態可用於允許相同程式碼二元與不同的使用情況執行，不同的使用情況可具有對於優先級安全及/或處理效能之不同偏好。

【0016】 廣泛範圍的處理操作可分別實施作為第一操作及第二操作。

【0017】 然而，NOP相容指令可特別有用於與功能呼叫相關聯的功能導言操作。因此，對於NOP相容指令之功能導言變體，第一操作及第二操作係與功能呼叫相關聯的功能導言操作。例如，功能導言操作可係在進入功能主體之前，在做出功能呼叫之前、期間或不久之後執行的初步操作。因為相同功能可在給定軟體工作負載執行期間被呼叫大量的次數，即使在呼叫功能之單一實例上達成相對較小的效能節省也可導致整體軟體工作負載效能的大幅改善，因為該改善係在功能每次被呼叫時都可看到。因

此，藉由使與功能呼叫相關聯的第一操作及第二操作能夠回應於單一NOP相容指令而執行，而非需要多個指令，此對於整體工作負載可提供明顯的效能改善。

【0018】 對於類似原因，NOP相容指令可用於與來自功能之處理的返回相關聯的功能收尾操作。此等可係在功能之主要主體完成之後執行的操作，以準備返回到呼叫該功能之背景處理。此類收尾操作可在實際將處理返回到背景處理的返回分支之前、期間或之後執行。因此，對於NOP相容指令之功能收尾變體，第一操作及第二操作係與來自功能之處理的返回相關聯的功能收尾操作。

【0019】 在一些實例中，對於該NOP相容指令之至少一變體，該第一操作及該第二操作中之一者包含一認證碼產生操作，以基於一運算元而產生一認證碼，且將該認證碼與該運算元相關聯。

【0020】 此一認證碼產生操作可施加至任何運算元，但可特別有用，其中其施加至功能返回位址，該功能返回位址係設定在功能呼叫上，以表示一旦功能主體已完成時後續返回分支應返回處理的指令位址，以提供針對返回導向程式化(Return-oriented-programming, ROP)攻擊的防禦。

【0021】 ROP型攻擊是資料處理系統上常見的攻擊類別。ROP攻擊是藉由損壞用以自功能呼叫或例外返回的返回狀態資訊而嘗試使程式以非預期方式動作的攻擊。通常，軟體將將返回狀態資訊儲存至記憶體，例如，以促進功能呼叫或例外的嵌套。在可以用於內部功能呼叫或例外的返回狀態資訊在暫存器中覆寫用於（功能呼叫或例外的嵌套集合的）外部功能呼叫或例外的返回狀態資訊之前，用於外部功能呼叫或例外的返回狀態

資訊可儲存至記憶體以保存其。ROP攻擊可在返回狀態資訊被恢復至暫存器且用以控制功能返回或例外返回之前，在返回狀態資訊儲存於記憶體中時，嘗試竄改返回狀態資訊。成功的ROP攻擊可導致功能返回或例外返回將程式流程返回至功能被呼叫或例外被採取的點之後的下一指令之外的一指令，這可允許攻擊者控制處理電路系統以執行程式設計者所意欲的操作序列以外的任意操作。

【0022】 認證碼產生操作可幫助保護免於此類ROP攻擊，藉由：產生對應於運算元（例如，功能返回位址）的認證碼，使得竄改儲存在記憶體中的運算元之後續嘗試可基於經竄改運算元及對應的認證碼之間的不匹配而刪除，若運算元已經竄改，認證碼可能不再對應於運算元。雖然認證碼產生操作可用於安全性，但其不是必要的，且在一些使用情況下，針對效能原因，較佳地省略認證碼產生操作。因此，可為有用的是提供NOP相容指令，其實現回應於指令而選擇是否執行認證碼產生操作，使得針對是否執行認證碼產生操作，相同程式碼序列可在不同的情境中執行而有不同的結果。因此，可有用的是第一操作及第二操作中之一者包含認證碼產生操作。

【0023】 雖然用於認證碼產生操作的運算元可係任意運算元（例如，自NOP相容指令指定的暫存器獲得的運算元），在一些實例中，該運算元包含獲自一連結暫存器的一值。回應於一功能返回分支指令，該指令解碼器可控制該處理電路系統以分支至該連結暫存器中指定的一位址。因此，當施加認證碼產生操作至連結暫存器中之運算元時，通常，用於認證碼產生操作的運算元可係功能返回位址。此可有用於提供防禦抵抗ROP攻

擊。在認證碼產生操作施加至功能返回位址之情況下，則此可係如上文所論述之功能導言操作的實例，因為其通常是此操作將相關於功能呼叫執行的情況。

【0024】 在認證碼產生操作中，所產生的認證碼可以不同的方式與運算元相關聯。例如，認證碼可儲存至具有與提供運算元之暫存器的已知關聯的特定暫存器。然而，此可非必要，且在一些情況下，認證碼可嵌入在運算元本身之部分中。因此，將該認證碼與該運算元相關聯可包含將該認證碼嵌入在該運算元之更高有效位元的一部分中。此可係有用的，因為藉由在運算元本身中嵌入認證碼，則此意指任何後續操作以移動運算元從一個位置至另一個位置（例如，將運算元自一暫存器推入至記憶體中的堆疊資料結構上）亦隱含地導致認證碼與運算元一起被轉移，而不需要單獨操作以轉移認證碼。運算元之更有效位元可用於表示認證碼，因為通常，雖然實務上處理器架構可支援具有某個數目的位元（例如，64位元）的位址，真實世界資料處理裝置可能還不需提供使用整個64位元位址空間之記憶體儲存。因此，雖然位址可具有64位元，但實務上僅使用較小數目的位元，其中數個最高有效位元對應於零（或一些其他固定值）。因此，因為實務上未使用一些上位元，這些位元可以認證碼取代（認證碼可插入至位址的上端處的這些未使用位元的任何子集中）。

【0025】 該認證碼產生操作可包含至少基於該運算元及一密鑰根據一密碼功能而產生該認證碼。藉由根據密碼安全功能（諸如QARMA-64、QARMA-128或SHA256）產生認證碼，例如基於密鑰，可以攻擊器難以預測對應於給定位址的認證碼之方式產生認證碼。

【0026】 在一些實例中，認證碼亦可取決於至密碼功能的修飾符輸入。修飾符可例如係與處理的目前點相關聯的一值，諸如一堆疊指標的一目前值。此可幫助防止重新使用攻擊，其中攻擊器獲得在程式之一點處使用的有效運算元認證碼對，且嘗試取代該運算元以用於在程式之不同點處使用的不同運算元。

【0027】 在一些實例中，對於該NOP相容指令之至少一變體，該第一操作及該第二操作中之一者包含一保護控制堆疊(guarded-control-stack, GCS)推送操作，以將該運算元推送至用於保護返回狀態資訊的一GCS資料結構。此類GCS推送操作係抵抗ROP攻擊之防禦措施的另一實例，但非依賴指派認證碼以保護返回狀態免於竄改，可建立保護的GCS資料結構，其具有至少一防禦措施以限制在GCS資料結構中寫入資料的能力，提供相對於正常記憶體區域之一些額外保護。再次，GCS推送操作可係功能導言操作的實例，因為其可用於在呼叫功能時在功能返回位址上執行GCS推送操作。雖然此類GCS推送操作可用於安全性，但其具有效能成本，且因此，具有較低安全要求的一些使用情況可能偏向不執行其。因此，GCS推送操作係一操作的另一實例，其可使用NOP相容指令有效地實施，使得包括NOP相容指令的相同程式碼序列可在不同使用情況中執行，其中指令功能選擇資訊控制GCS推送操作是否實際實行。

【0028】 該NOP相容指令特別有用於該NOP相容指令之變體，其中該第一操作及該第二操作中之一者包含一認證碼產生操作，以基於一運算元而產生一認證碼，且將該認證碼與該運算元相關聯；及該第一操作及該第二操作中之另一者包含一保護控制堆疊(GCS)推送操作，以將該運算元

推送至用於保護返回狀態資訊的一GCS資料結構。由於認證碼產生操作及GCS推送操作可視為用於保護功能返回狀態免於ROP攻擊的替代技術，在相同程式（其與功能呼叫相關聯）點處通常需要彼等，且因此有用的是將彼等組合成單一指令，同時亦提供關閉此等操作之一或兩者的選項。雖然兩個操作名義上保護免於相同類別的攻擊，但其等可具有不同的利弊，且因此針對「防禦深度」，一些開發人員可希望包括兩個措施，使得其可用於支援在一功能呼叫下執行兩個操作的選項。藉由使用NOP相容指令，可僅執行單一指令以執行兩種類型之操作（在指令功能選擇資訊在第二狀態中的情況中）。

【0029】　在第一操作及第二操作分別係認證碼產生操作及GCS推送操作的情況下（或反之亦然），在此等操作之間可能有不同的順序。一些實施方案可因此允許選擇使用哪個順序，取決於指令功能選擇資訊。

【0030】　回應於該NOP相容指令，當指令功能選擇資訊係在該第二狀態之一第一子狀態中時，該處理電路系統可將該運算元及該認證碼兩者推送至該GCS資料結構（例如，此可對應於先執行該認證碼產生操作，以在該運算元中嵌入該認證碼，且接著在該認證碼產生操作的結果上執行GCS推送操作）。此方法可藉由使用GCS資料結構來保護認證碼而改善安全性。

【0031】　回應於該NOP相容指令，當該指令功能選擇資訊係在該第二狀態之一第二子狀態中時，該處理電路系統可推送該運算元、但不推送該認證碼至該GCS資料結構。在此情況下，認證碼產生操作及GCS推送操作

可彼此獨立，且因此可並行或以任一順序執行。藉由支援並行地執行其等之選項，此可改善效能，但意指GCS資料結構不保護認證碼。

【0032】 在另一實例中，對於NOP相容指令之至少一變體，該第一操作及該第二操作中之一者包含一認證碼檢查操作，以檢查與一運算元相關聯的一相關聯認證碼是否匹配於基於該運算元產生的一預期認證碼，且回應於偵測到該相關聯認證碼及該預期認證碼之間的一失配而觸發一錯誤處理回應。此操作可用以檢查由上文所描述之認證碼產生操作所產生之認證碼的有效性，且儘管其可在任何運算元上執行，但其通常可執行作為功能收尾操作，以檢查返回位址是否安全以使用（如上文所提及之防禦ROP攻擊）。因此，針對認證碼產生操作的對應原因，NOP相容指令可用於認證碼檢查操作。

【0033】 在一些實例中，該認證碼檢查操作的相關聯的認證碼可自運算元的較高有效位元的一部分獲得。

【0034】 在對於認證碼產生操作之對應的方式中，在認證碼檢查操作中，預期認證碼可至少基於該運算元及一密鑰（且在一些情況下亦基於修飾符（諸如，堆疊指標））根據密碼功能而產生。

【0035】 對於該NOP相容指令之至少一變體，該第一操作及該第二操作中之一者可包含一保護控制堆疊(GCS)取出操作，以自用於保護該功能返回資訊的一GCS資料結構取出功能返回資訊。此可用以獲得返回狀態資訊，該返回狀態資訊先前被先前的GCS推送操作推送至GCS資料結構。對於GCS推送操作有類似原因，GCS取出操作（功能收尾操作的實例）可用於使用NOP相容指令實施。

【0036】 再次，NOP相容指令的一些變體可支援認證碼檢查操作及GCS取出操作兩者。因此，該第一操作及該第二操作中之一者可包含一認證碼檢查操作，以檢查與一運算元相關聯的一相關聯認證碼是否匹配於基於該運算元產生的一預期認證碼，且回應於偵測到該相關聯認證碼及該預期認證碼之間的一失配而觸發一錯誤處理回應；及該第一操作及該第二操作中之另一者包含一保護控制堆疊(GCS)取出操作，以自用於保護該功能返回資訊的一GCS資料結構取出功能返回資訊。此等操作可用於組合至相同的NOP相容指令中，因為若需要兩者，其等將通常在程式中的相同點處執行，如同在執行功能返回之前、在功能的主要主體的完成之後的功能收尾操作。

【0037】 可基於指令功能選擇資訊控制認證碼檢查操作及GCS取出操作之間的順序。回應於該NOP相容指令，當該指令功能選擇資訊係在該第二狀態之一第一子狀態中時，該處理電路系統可執行該GCS取出操作，並執行該認證碼檢查操作於該GCS取出操作自該GCS資料結構取出的一值；及回應於該NOP相容指令，當該指令功能選擇資訊係在該第二狀態之一第二子狀態中時，該處理電路系統可在執行該GCS取出操作之前執行該認證碼檢查操作於一給定暫存器中的一值，並執行該GCS取出操作，以自該GCS資料結構取出該功能返回資訊至該給定暫存器。再次，此提供針對效能與安全的折衷之不同選項。

【0038】 對於NOP相容指令實施GCS推送操作或GCS取出操作作為第一/第二操作之一者的情況，則回應於NOP相容指令，當該指令功能選擇資訊係在指示對於GCS資料結構的存取係回應於NOP相容指令而執行之狀態

中（亦即，當將執行GCS推送操作及GCS取出操作之一者時），該處理電路系統可回應於偵測到對應於該NOP相容指令之一目標位址的一記憶體區域係由記憶體屬性資料指定為用於儲存該GCS資料結構的GCS區域之外的一記憶體區域，拒絕NOP相容指令所觸發的記憶體存取。因此，若存取係由GCS存取類型之指令觸發，則可拒絕對於未指定為GCS資料結構的記憶體區域的存取。此避免GCS存取類型指令（當NOP相容指令執行GCS推送操作或GCS取出操作時，包括NOP相容指令）被誤用而存取不意欲用於儲存GCS資料結構的記憶體區域，其可減少攻擊器可利用的攻擊面。

【0039】 類似地，該處理電路系統可回應於偵測到對應於該非GCS存取類型指令之一目標位址的一記憶體區域係由記憶體屬性資料指定為該GCS區域，拒絕一非GCS存取類型指令所觸發的一寫入記憶體存取。藉由將寫入至GCS區域的能力限制至GCS存取類型指令（當指令功能選擇資訊係在指示將執行GCS存取之狀態中時，包括NOP相容指令），其他較一般的記憶體存取指令不能竄改GCS資料結構的內容，提供對於儲存在GCS資料結構中的保護的返回狀態資訊更大的安全保證。再次，此減少攻擊器嘗試安裝ROP攻擊時可利用之攻擊面。

【0040】 指令功能選擇資訊可以不同方式表示。在一些實例中，稍早提及之第一、第二及第三狀態對應於指令功能選擇資訊之不同（可能任意選擇之）編碼。因此，可使用在第一、第二及第三狀態與指令功能選擇資訊之位元值的不同組合之間的任何映射。

【0041】 然而，在一些實例中，可有用的是，該指令功能選擇資訊包含一第一操作指示符，以指示該第一操作是否將回應於該NOP相容指令

被執行；及一第二操作指示符，以指示該第二操作是否將回應於該NOP相容指令被執行。例如，指令功能選擇資訊可包含一組位元，其中各位元對應於可回應於NOP相容指令可能被選擇執行的操作中之一者，且指示是否需要執行該操作。指令功能選擇資訊的此編碼可容易由軟體開發者理解且較簡單由處理電路系統的硬體或指令解碼器解碼，因為各操作的選擇僅取決於單一指示符（例如，單一位元）而非需要更複雜之解碼電路邏輯。

【0042】　在一些實例中，基於指令功能選擇資訊，NOP相容指令可支援自超過兩個操作選擇之選項。因此，該指令功能選擇資訊可進一步指示該處理電路系統是否應回應於該NOP相容指令而執行一第三操作。例如，其中指令功能選擇資訊包含一組位元，各指示是否應回應於該NOP相容指令而執行各別操作，可相對有效率的是增加對於所欲的額外操作的支援。因此，雖然下文討論的實例展示具有兩個操作之實例，申請專利範圍不限於此，且亦可支援額外操作。

【0043】　上文討論的技術可實施在資料處理設備內，該資料處理設備具有針對實施上文討論之處理電路系統及指令解碼器而提供的硬體電路系統。

【0044】　然而，相同技術亦可實施在電腦程式內，該電腦程式係在主機資料處理設備上執行以提供用於目標碼之執行的指令執行環境。即使主機資料處理設備本身不支援該架構，此一電腦程式可控制主機資料處理設備以模擬其將提供在實際支援根據某個指令集架構之目標碼的硬體設備上的架構環境。電腦程式可具有指令解碼程式邏輯及暫存器仿真程式邏輯，其控制主機資料處理設備以仿真上文所討論之特徵，包括針對如上文

所描述之NOP相容指令的支援。指令解碼程式邏輯解碼目標碼的指令，並產生由主機支援的原生架構的指令，以仿真由目標碼中經解碼指令表示的功能。暫存器仿真程式邏輯將資料維持在該主機資料處理設備的儲存電路系統中，以仿真至少一控制暫存器的內容，包括儲存如上文所論述之指令功能選擇資訊的（多個）暫存器。因此，當包括NOP相容指令的目標碼在由主機資料處理設備上執行的模擬電腦程式所提供的指令執行環境中執行時，即使主機資料處理設備本身不支援NOP相容指令，也可達成如上文所論述之相同功能。

【0045】　例如，當針對一個指令集架構所編寫之程式碼係在支援一不同指令集架構的一主機處理器上執行時，此一模擬程式可係有用的。再者，由於軟體在模擬執行環境上的執行可使軟體測試得以與支援新架構之硬體裝置的進行中開發平行，模擬可允許針對一較新版本的指令集架構的軟體開發在支援該新架構版本的處理硬體就緒之前開始。模擬程式可儲存在儲存媒體上，該儲存媒體可係非暫時性儲存媒體。

【0046】　圖1示意地繪示資料處理設備2的一實例。資料處理設備具有包括若干個管線級的處理管線4。在此實例中，管線級包括用於從指令快取記憶體8提取指令的提取級6；用於解碼提取程式指令之解碼級10，以產生待由管線的其餘級處理的微操作(micro-operation)（經解碼指令）；用於檢查微操作所需的運算元在暫存器檔案14中是否可用，並發布用於將給定微操作的所需運算元執行一次的微操作係可用的發布級12；用於執行對應於藉由處理讀自暫存器檔案14之運算元以產生結果值之微操作的資料處理操作的執行級16；及用於將處理的結果寫回至暫存器檔案14的寫回級

18。應瞭解此僅係可能的管線架構的一個實例，且其他系統可具有額外級或不同的級組態。例如，在亂序處理器中，可包括用於將由程式指令或微操作指定的架構暫存器映射至識別暫存器檔案14中之實體暫存器之實體暫存器說明符的暫存器重命名級。在一些實例中，在由解碼級10解碼的程式指令與由執行級所處理之對應微操作之間可有一對一關係。在程式指令與微操作之間亦係可能有一對多或多對一關係，使得，例如，單一程式指令可分成二或更多個微操作，或者二或更多個程式指令可融合以作為單一微操作來處理。

【0047】 執行級16（處理電路系統之實例）包括若干個處理單元，以用於執行不同類別的處理操作。例如，執行單元可包括用於對讀自暫存器14的純量運算元進行算術或邏輯運算的純量算術/邏輯單元(arithmetic/logic unit, ALU) 20；用於對浮點值進行運算的浮點單元22；用於評估分支操作之結果及調整該程式計數器的分支單元24，該程式化計數器據此表示目前的執行點；及用於執行載入/儲存操作以存取記憶體系統8、30、32、34中之資料的載入/儲存單元26。提供一記憶體管理單元(memory management unit, MMU) 28，其係記憶體管理電路系統之實例，以用於執行由載入/儲存單元26基於資料存取指令之運算元所指定的虛擬位址與識別記憶體系統中資料之儲存位置的實體位址之間的位址轉譯。MMU具有一轉譯後備緩衝區(translation lookaside buffer, TLB) 29，其用於從儲存在記憶體系統中的分頁表快取位址轉譯資料，其中分頁表的分頁表項目定義位址轉譯映射及存取權限，其（例如）管控是否允許管線上執行的給定程序從給定記憶體區域讀取或寫入資料或執行指令。當分頁表結構

被遍歷以定位對應於所需位址的分頁表項目時，MMU 28可具有在分頁表走訪期間請求記憶體存取的電路系統。

【0048】 在此實例中，記憶體系統包括一級資料快取記憶體30、一級指令快取記憶體8、共用二級快取記憶體32、及主系統記憶體34。將理解此僅係可能的記憶體階層的一個實例，並可提供快取記憶體的其他配置。顯示在執行級16中的處理單元20至26的具體類型僅係一個實例，且其他實施方案可具有不同組的處理單元或可包括相同類型的處理單元的多個實例，使得可平行地處理多個相同類型的微操作。應理解圖1僅係一可能的處理器管線實施方案的一些組件的簡化表示，且處理器可包括為了簡潔起見而未繪示的許多其他元件。雖然圖1顯示具有對記憶體34存取的單一處理器核心，該設備2亦可具有一或多個另外的處理器核心，其共享對記憶體34的存取，其中各核心具有各別快取記憶體8、30、32。

【0049】 圖2繪示設備2之一些暫存器14的實例。應瞭解，圖2未顯示所有暫存器-設備亦可包括其他暫存器。提供一組通用暫存器50，用於儲存通用運算元及處理操作之結果。此等通用暫存器中之一些亦可具有更特定功能，諸如用於儲存功能返回位址之連結暫存器(link register, LR)，其可使用通用暫存器識別符中之一者（例如，暫存器X30）來定址。暫存器亦包括用於儲存堆疊指標的堆疊指標(stack pointer, SP)暫存器52。該設備亦具有一些控制暫存器56，其用於儲存用以控制該處理器之操作的控制資訊。例如，控制暫存器56可包括保護控制堆疊(GCS)堆疊指標暫存器58，其用以控制對該GCS的存取，如下文相關於圖6進一步討論的；及至少一暫存器，其提供用以控制如下文進一步論述之NOP相容指令的行為之

指令功能選擇資訊60。雖然圖2顯示指令功能選擇資訊60作為單一暫存器，但此資訊亦可能分割至二或更多個暫存器。

【0050】 圖3展示NOP相容指令70之實例。指令編碼包括識別指令類型的運算碼72、及用於指定指令的運算元之一或多個運算元欄位74、76。運算元74、76可使用立即值、或使用指定儲存運算元的暫存器14之暫存器識別符、或至少一立即值及至少一暫存器識別符兩者之組合來指定。在一些情況下，該指令亦可指定識別待寫入指令之結果的暫存器的額外目的地欄位，或替代地針對亦可充當目的地暫存器的運算元欄位中之一者所指定的暫存器的一者。雖然圖3為了實例的緣故顯示了兩個運算元，但NOP相容指令的其他實例可具有較大或較小數量的運算元。

【0051】 在具有相同運算碼72及運算元欄位74、76的相同定義之單一指令編碼內，NOP相容指令代表完全不執行操作（使得該指令之動作如同NOP指令）或執行至少兩個不同處理操作（包括至少第一操作及第二操作）的一或多者之選項。基於儲存在控制暫存器56中的指令功能選擇資訊60，回應於NOP相容指令經選擇而執行操作的哪個組合（藉由指令解碼器10及/或處理管線4的執行級16）。此資訊可藉由處理器所執行的指令來設定。例如，系統暫存器修改指令（其可限制成在某些執行狀態或優先層級中執行）可用以設定指令功能選擇資訊60。替代地，可存在用於設定指令功能選擇資訊的專用類型指令，其不同於用於修改其他控制暫存器56的系統暫存器修改指令。在執行包括NOP相容指令之一段程式碼的使用情況下，一些實例可使用相同一段程式碼的較早指令來設定指令功能選擇資訊60。在其他實例中，指令功能選擇資訊60可由監督程式碼設定，其管理包

括NOP相容指令之程式碼之執行（例如，藉由管理應用程式的執行之作業系統，或管理作業系統的執行之監督器）。

【0052】 圖4係繪示處理NOP相容指令之方法的流程圖。在步驟100，指令解碼電路系統10判定NOP相容指令是否經解碼。若否，則指令解碼電路系統10解碼另一類型的指令，控制處理器以執行由該指令表示的操作，並繼續等待經解碼的NOP相容指令。

【0053】 當NOP相容指令經解碼，則在步驟102處，指令解碼器10檢查（或控制處理器之另一部分，諸如執行級16，以檢查）儲存在控制暫存器56中的指令功能選擇資訊60的狀態。

【0054】 若指令功能選擇資訊60在第一狀態中，則在步驟104處，指令解碼器10控制執行級16以將NOP相容指令視為NOP指令。因此，回應於NOP相容指令，處理電路系統不會導致架構狀態的任何變化（除了前進程式計數器以指向NOP指令之後的下一依序指令）。

【0055】 若指令功能選擇資訊60在第二狀態中，則在步驟106處，處理取決於指令功能選擇資訊是否是第二狀態的第一子狀態或第二子狀態。若指令功能選擇資訊60係第二狀態的第一子狀態，則在步驟108處，處理電路系統16經控制以執行第一操作及第二操作兩者（以該第一操作與第二操作之間的第一順序）。若指令功能選擇資訊60係第二狀態的第二子狀態，則在步驟110處，處理電路系統16經控制以執行第一操作及第二操作兩者（以該第一操作與第二操作之間的第二順序，其不同於第一順序）。第一及第二順序可不同在於第一及第二操作是否依序或並行執行，或不同在於第一及第二操作哪個第一執行且哪個第二執行。順序亦可不同

在於第一操作與第二操作之間是否有任何相依性（亦即，是否一個操作取決於另一者的結果或兩個操作是否獨立）。

【0056】 用於控制第一操作與第二操作之間的順序的支援係可選的，且在一些實例中，步驟106及110可省略，使得當指令功能選擇資訊係在第二狀態時，則該方法繼續進行至步驟108，以用預設選擇的第一順序執行第一操作及第二操作。

【0057】 下文相關於圖9至圖12描述控制操作之間的順序的特定實例。

【0058】 在步驟102處，若指令功能選擇資訊60係在第三狀態，則在步驟112處，控制處理電路系統16以執行第一操作但不執行第二操作。

【0059】 若指令功能選擇資訊係在第四狀態，則在步驟114處，控制處理電路系統16以執行第二操作但不執行第一操作。支援步驟140可係可選的，且在一些實例中，NOP相容指令不可能執行第二操作但不執行第一操作。例如，在一些使用情況中，若用於指令功能選擇資訊的四個不同狀態有唯一空間（例如，因為僅2位元用於此資訊60），則一些實施方案可偏好使用指令功能選擇資訊60的第四編碼，以允許第二狀態的二個不同子狀態，如步驟108及110所示，使得操作的順序可受控制。其他實例可支援圖3之所有狀態及子狀態，且因此可使用具有3或更多位元之指令功能選擇資訊以允許指令功能選擇資訊的這些額外編碼。

【0060】 指令功能選擇資訊的一個可能編碼（不具有用於控制第一操作與第二操作之間的順序的支援）可如下：

- 0b00-第一操作停用，第二操作停用-指令動作如 NOP（第一狀態）；
- 0b01-第一操作啟用，第二操作停用（第三狀態）；
- 0b10-第一操作停用，第二操作啟用（第四狀態）；
- 0b11-第一操作啟用，第二操作啟用（第二狀態）。

【0061】 此一編碼可針對各操作提供不同的位元，其等獨立地指示是否啟用或停用該操作。藉由提供每個操作額外位元（各自啟用/關閉任何額外操作），此可延伸至第三操作或進一步操作。

【0062】 指令功能選擇資訊的另一實例編碼分配有執行第二操作而不執行第一操作的能力，但使用指令功能選擇資訊60的第四編碼，以在執行兩個操作時指示第一操作與第二操作之間的所欲順序：

- 0b00-第一操作停用，第二操作停用-指令動作如 NOP（第一狀態）；
- 0b01-第一操作啟用，第二操作停用（第三狀態）；
- 0b10-第一操作啟用，第二操作啟用，第一操作及第二操作之間的第一順序（第二狀態-第一子狀態）；
- 0b11-第一操作啟用，第二操作啟用，第一操作及第二操作之間的第二順序（第二狀態-第二子狀態）；

【0063】 另一實例編碼可使用超過2個位元且留下一些編碼以備未來使用，例如，在加入額外操作或組態選項的支援時：

- 0bX00-第一操作停用，第二操作停用-指令動作如 NOP（第一狀態）；

- 0bX01-第一操作啟用，第二操作停用（第三狀態）；
- 0bX10-第一操作停用，第二操作啟用（第四狀態）；
- 0b011-第一操作啟用，第二操作啟用，第一操作及第二操作之間的第一順序（第二狀態-第一子狀態）；
- 0b111-第一操作啟用，第二操作啟用，第一操作及第二操作之間的第二順序（第二狀態-第二子狀態）；

【0064】 應瞭解，所有此等實例僅係一些方式，其中針對NOP指令將執行的操作可由指令功能選擇資訊60編碼。

【0065】 各種處理操作可用作第一及第二操作。然而，一個特定使用情形可用於功能導言或收尾操作分別在功能呼叫或功能返回上執行時。具體而言，可有用的是，第一操作及第二操作係在功能返回狀態上用於保護免於返回導向程式化(return oriented programming, ROP)攻擊的替代操作。例如，第一操作及第二操作可係下文進一步描述的指標認證/檢查操作或GCS推送/取出操作。

【0066】 圖5繪示呼叫功能（為了易於參見，標示為fn1）及自功能返回的實例。功能（亦稱為程序）係可自程式之另一部分呼叫的指令序列，且當完成時其將處理返回至該功能被呼叫之程式流程之部分。相同功能可自程式中之多個不同位置被呼叫，且因此在呼叫功能時儲存功能返回位址，使得功能返回可區分程式流程應返回至哪個位址。

【0067】 例如，如圖5所示，具有連結指令BLR的分支可在功能將被呼叫的該點（由位址#add1表示）處執行，以使程式流程分支至使用具有連結指令的分支的運算元所指定之分支目標位址#add2處的指令。具有連

結指令之分支亦導致處理電路系統設定連結暫存器（用於追蹤功能返回位址之指定暫存器，例如，如上文所示之通用暫存器）至在具有連結指令之分支之後的下一指令的位址（在此實例中，功能返回位址係#add+4）。在已採取分支之後，在功能碼中執行數個指令（例如，LD、MUL、ADD等），且當功能完成時，執行返回分支指令RET，其使分支至儲存在連結暫存器中的返回位址指示的指令。

【0068】 若無其他功能自fn1內被呼叫，且在到達fn1之結束處的返回分支之前無例外發生，則在連結暫存器中的位址仍應相同如fn1被呼叫時所設定的一般。

【0069】 然而，通常，由背景程式碼所呼叫的第一功能fn1本身可以嵌套方式呼叫另一功能（就是說fn2），且在此情況中，對fn2之功能呼叫將覆寫儲存在連結暫存器中的返回位址，且因此在呼叫另一功能之前，第一功能fn1的功能程式碼應包括一指令，以將來自連結暫存器之返回位址儲存至記憶體中的資料結構（例如，堆疊結構，以後進先出(last-in-first-out, LIFO)方式操作），且在自fn2返回之後，fn1之功能程式碼應將返回位址恢復至執行返回分支之前的連結暫存器。用於儲存及恢復功能返回狀態（諸如返回位址）之職責將通常在於軟體（可不存在用於儲存返回位址之架構上強制硬體機構）。

【0070】 然而，當功能返回位址儲存在記憶體中時，可能易於受到攻擊器修改該資料之攻擊，例如使用在另一處理器核心上執行的另一執行緒，或藉由中斷該呼叫功能且同時執行其他程式碼，其覆寫儲存在記憶體中的該返回位址。替代地，該攻擊器可執行一些指令，其目標在於修改將

該返回位址自記憶體恢復至暫存器之該指令之位址運算元，使得從記憶體載入的該資料不同於在呼叫嵌套功能之前原本儲存至記憶體的返回位址。若攻擊器可導致返回分支分支至程式流程中並非功能呼叫分支之後的指令之點，攻擊器可能夠導致軟體不正確動作，且可能夠繞過某些安全保護或導致執行非所欲操作。

【0071】 功能呼叫係操作的一實例，其產生返回狀態資訊，返回狀態資訊提供關於處理電路系統稍後將被恢復至的狀態的資訊。返回狀態資訊可被擷取之另一情境可係採取例外時，其中提供在硬體中的例外處理電路系統，或軟體例外處理器可擷取例外返回狀態資訊，諸如指示在從處理一例外返回之後將執行之指令的位址的例外返回位址，及/或指示在從例外返回之後處理器將執行的模式或執行狀態之儲存的處理器狀態資訊。例如，儲存的處理器狀態資訊可指示採取例外的例外層級，以及關於採取例外時處理器的操作狀態的其他資訊。如同功能呼叫，例外可被嵌套，且因此當採取另一例外時，針對一例外擷取的例外返回狀態可儲存至記憶體（以硬體自動地，或藉由軟體例外處理器），且因此當其儲存在儲存記憶體中時可能容易受攻擊器竄改。這些類型的攻擊可稱為返回導向程式化(ROP)攻擊。可能期望提供針對此類攻擊的架構反制措施。

【0072】 圖6繪示用於使用在記憶體中稱為「保護控制堆疊(guarded control stack, GCS)」之保護的資料結構120保護免於ROP攻擊的方法。在記憶體位址空間內之GCS資料結構的位置可藉由軟體選擇，但硬體提供經設計以保護GCS資料結構免於受惡意攻擊者竄改之架構特徵。

【0073】 如圖2所示，暫存器14包括控制暫存器56，其包括用於儲存一堆疊指標的一或多個經保護控制堆疊指標（GCS指標）暫存器58，該堆疊指標指示GCS資料結構上的位址。在一些實例中，GCS指標暫存器58可係堆積的暫存器組，分別提供用於至少兩個執行狀態（例如，例外層級），以實現在不同執行狀態下操作的軟體參照記憶體內之不同GCS結構，而不需要在執行狀態之各轉變之後重新程式化共用的堆疊指標暫存器。其他實例可使用單一GCS指標暫存器，且軟體可在執行狀態之間的轉變時更新儲存在GCS指標暫存器58中的堆疊指標。

【0074】 如圖6所示，GCS資料結構120儲存在由用於控制位址轉譯及存取權限檢查的記憶體管理單元(MMU) 28所使用之分頁表的相關聯分頁表項目直接或間接指定的記憶體屬性所指定為記憶體之GCS區域的記憶體區域中。GCS區域屬性可直接在對應分頁表項目之編碼內指定，或可間接參照在由分頁表項目所參照之暫存器內。

【0075】 當記憶體區域被識別為GCS區域時，則當執行GCS存取指令的一特定子集時，至該區域的寫入存取係限於在由處理電路系統16觸發的寫入請求。由軟體使用用於一般儲存操作而不意欲存取GCS結構的通用儲存指令並不視為GCS存取指令之限制子集中之一者。MMU 28仍可允許使用通用載入指令讀取GCS結構，該通用載入指令導致發佈並非GCS記憶體存取請求的讀取請求。當記憶體存取請求請求對GCS區域之存取時，該請求係寫入請求，且該請求不是由GCS存取指令之限制子集的一者觸發的GCS記憶體存取請求，則該記憶體存取請求被拒絕且傳訊錯誤。GCS存取指令的子集可包括至少一GCS推送指令，其導致返回狀態資訊（諸如來自

連結暫存器的功能返回位址，或例外返回位址或採取例外時被擷取之經儲存的處理器狀態）被推送至使用GCS指標暫存器58中指示的堆疊指標所判定的GCS結構上的位置。GCS存取指令亦可包括至少一種形式之GCS取出指令，其自GCS結構取出保護的返回資訊。

【0076】　相反地，GCS存取指令可不被允許存取未由分頁表屬性指定為GCS區域類型之記憶體區域。因此，當由該存取之目標記憶體區域未標記為GCS區域類型時，若嘗試執行GCS存取，則可傳訊錯誤。藉由禁止使用GCS存取指令存取非GCS區域，此不鼓勵程式設計者使用GCS存取指令，除非真的意欲是GCS存取，以減少可用於攻擊器的攻擊面。

【0077】　GCS結構與由軟體使用的任何資料結構分開，以將儲存的返回狀態資訊維持在記憶體中，以處理功能呼叫或例外的嵌套。因此，當功能呼叫或例外嵌套時，GCS結構不意欲消除軟體本身追蹤返回狀態資訊的儲存及恢復之需求（返回狀態的軟體觸發儲存可以如在不支援上文所討論的GCS保護之架構措施的處理器上相同的方式繼續）。替代地，GCS結構提供經保護記憶體的區域，其被保護免於危害的程式碼的竄改，其可用以提供用於驗證意欲由軟體使用以從功能呼叫或例外的處理返回之返回狀態資訊的資訊。

【0078】　在一些實施方案中，使經保護返回狀態資訊從GCS結構取出的GCS取出指令亦可導致處理電路系統16比較取出的返回狀態與儲存在暫存器中的目前返回狀態資訊（例如，用於功能返回的連結暫存器54，或例外返回位址暫存器及/或用於例外返回之經儲存的處理器狀態暫存器），且若從GCS結構120取出的返回狀態資訊與軟體意欲用於功能/異常返回的

預期返回狀態資訊之間存在失配，則傳訊錯誤。因此，軟體可藉由包括功能呼叫/返回或例外進入/返回時執行的程式碼內之GCS推送及GCS取出指令之實例而受保護免於竄改。

【0079】 其他實施方案可界定用於驗證意欲的返回狀態資訊是否有效的單獨指令，不同於從GCS結構120取出返回狀態資訊的指令。

【0080】 替代地，GCS取出指令可直接將來自GCS的經保護返回狀態取出至用以指定例外返回或功能返回的返回狀態之一或多個暫存器（或可與例外/功能返回指令組合，以取出經保護返回狀態並且使用該狀態用於控制例外/功能返回），在此情況中，不必實行驗證是否軟體提供的意欲的返回狀態資訊係有效，因為在此一實施方案中，GCS保護的返回狀態係直接用以控制例外/功能返回。例如，對於功能返回位址的GCS保護，可將功能返回位址直接取出至連結暫存器54，取代軟體可基於其自己的經管理堆疊結構而置放在其中的任何軟體管理之功能返回位址。

【0081】 此外，亦可支援存取指令之其他類型之GCS。當啟用GCS模式時（控制暫存器56中的控制狀態可控制是否GCS模式啟用），一些指令（其在停用GCS的使用之模式中具有其他功能）在執行時，可導致處理電路系統16執行額外功能（諸如額外的GCS模式特定安全檢查）。

【0082】 一般而言，藉由提供用於定義用於GCS結構120之GCS記憶體區類型的架構支援，且限制對GCS區域類型的寫入存取至GCS存取指令的有限子集（其可能不被允許存取除了GCS區域類型以外的記憶體區域），此減少攻擊器可用於嘗試竄改儲存在GCS結構120上的保護的返回狀態資訊之攻擊面。

【0083】 GCS提供一種防禦抵抗ROP攻擊。用於保護免於ROP攻擊的另一選項是使用與返回狀態資訊相關聯的認證碼。圖7繪示基於第一源運算元src1回應於一認證碼產生指令而執行的一認證碼產生操作的實例。源運算元可係任何值，但其特別有用於施加認證碼產生操作至位址指標，諸如功能返回位址。源運算元可指定（例如，藉由參照來源暫存器，諸如連結暫存器）包含某個數目之位元X的位址，但實際上僅那些位元之某個數目Y可用於有效位址（例如，X可等於64且Y可等於48或52）。因此，位址的X-Y上位元可預設設定為零。

【0084】 在認證碼產生操作中，源運算元可傳遞至處理電路系統16的加密/解密電路系統（例如，執行級16可包括類似於圖1所示的其他執行單元20、22、24、26之加密/解密功能單元），其可基於來自密鑰儲存器的密鑰及至少一修飾符值，施加一認證碼產生功能140至第一來源值。將所得的認證碼(PAC)插入至指標位址的未使用上位元中，以產生指令的結果。結果可例如寫回至儲存源運算元的相同暫存器。例如，若與源運算元執行的是儲存在連結暫存器54中的目前功能返回位址，則結果寫回至連結暫存器54。認證碼產生功能40可使用密碼雜湊功能（例如，SHA256、SHA128、QARMA-128、QARMA-256等），其使其運算不可行以猜測與特定位址相關聯的認證碼，而不知道密鑰。修飾符值可係用於連接由該操作所產生的該認證碼的該特定實例至程式碼中到達的特定執行值之值，從而降低重新使用攻擊的風險，其中來自該程式碼之一部分的一有效位址/PAC對係不正確地被取代而在該程式碼之另一部分使用。例如，來自堆

疊指標暫存器52的堆疊指標、或代表所採取以到達目前處理點的功能呼叫史之呼叫路徑指示符，可用作修飾符。

【0085】 圖8顯示在第二源運算元src2上執行的對應認證碼檢查操作。第二源運算元預期是指標位址，其先前已藉由在圖7所示的認證碼產生操作中將認證碼PAC插入至其上位元而認證，但若攻擊器已經修改該指標，則認證碼(PAC)可能不有效。在認證碼檢查操作中，處理電路系統16使用對應於當由位址的上位元所代表的認證碼產生時預期已使用的密鑰及修飾符值之密鑰及修飾符值，將相同的認證碼產生功能140施加至第二源運算元的位址位元（排除認證碼PAC）。接著，將預期認證碼PAC與自第二源運算元src2的上位元提取的相關聯認證碼PAC比較，且判定是否預期認證碼與相關聯的認證碼匹配。若是，則允許處理繼續，而若預期及相關聯碼之間存在失配，則觸發錯誤處理回應，例如觸發一例外或將來源暫存器的上位元設定至對應於無效位址的一值，使得對於該位址的後續存取或指令提取將因為存取無效位址而觸發MMU 28觸發記憶體錯誤（這意指若具有不正確PAC之位址係用於功能返回，在返回分支之後後續嘗試自該位址提取指令將觸發錯誤，防止處理電路系統繼續執行不正確功能返回之外的程式）。

【0086】 藉由使用圖7及圖8之認證碼產生及檢查操作，此允許指標被認證，使得攻擊器更難以注入未認證之指標且成功使程式碼分支至該指標所識別之位置，保護抵抗ROP攻擊。藉由使用密碼功能作為認證碼產生功能140，此可使對與特定位址相關聯之認證碼的暴力猜測變困難。用於執行認證碼產生操作的指令可包括在當指標位址產生時的點處的程式碼中

（例如，在回應於功能呼叫設定連結暫存器及儲存自連結暫存器至記憶體的功能返回位址之間），且當稍後實際要使用該位址時（例如，在功能返回分支之前），可包括認證碼檢查指令AUT，以在實際分支至該位址之前，仔細檢查認證碼。

【0087】 認證碼產生功能140可隨實施方案變化，或可基於控制暫存器56中的控制狀態而在給定實施方案上可組態。用於認證碼產生功能的修飾符值亦可係可組態的或可能針對實施認證碼產生/檢查操作的指令的不同變體有所不同。

【0088】 上文關於圖6至圖8所描述之GCS操作及認證碼操作兩者皆可看作防禦抵抗ROP攻擊，但一些使用者可能偏好使用一種防禦形式，且其他使用者可使用另一形式。一些使用者可偏好使用兩者在深度方法中之防禦的組合。此外，在有時使用在需要ROP防禦之安全性的使用情況中之一段程式碼亦可在不需要此安全性的使用情況中執行的情境中，可能有時想要省略此等操作，此情況中可能較好的係針對效能而省略此等操作。因此，此等操作對於上文描述之NOP相容指令之第一及第二操作可係有用之實例。

【0089】 在一些情況下，可能期望提供程式二元，其向後相容於不支援GCS及認證碼特徵的舊硬體，且當在支援此等特徵的較新硬體上運行時支援新功能。因此，針對NOP相容指令選擇的運算碼72可係在較舊硬體上可被視為NOP指令的一者。

【0090】 通常，當新架構特徵添加至ISA時，控制暫存器可用以指示是否支援該特徵，且軟體在使用前可能需要檢查該特徵是否實施。對於

許多特徵，此係可接受的，但對於預期非常頻繁執行的特徵，諸如用於功能導言及收尾之特徵，由於效能的原因，此係不可行的。對於此類特徵，可有用的是提供執行無操作的一組NOP指令編碼，至少直到針對支援更新架構的較新硬體，一些功能被添加至該編碼。隨著更多特徵加入架構，可在對應於各額外特徵的功能導言及收尾中添加更多的NOP相容指令，但此將使功能更巨大，因為添加更多的NOP相容指令，引致快取結構及指令通量的成本，且因此降低效能。

【0091】 因此，使用上文描述之NOP相容指令，可將多個操作過載至單一NOP相容編碼上，其中提供在一或多個控制暫存器56中之指令功能選擇資訊60獨立地開啟/關閉各特徵。

【0092】 例如，如上文所論述，針對經保護控制堆疊特徵，其中吾人希望在功能進入時將連結暫存器之內容（例如，功能返回位址）推送至經保護堆疊上，且上文提及之認證碼產生操作（PAC特徵）簽署指標，諸如功能返回位址，此將通常涉及兩個指令，以執行GCS推送操作(GCSPUSH)及簽署操作(PACIASP)。類似地，在功能返回時，存在對應的指令，以執行GCS取出操作(GCS pop operation, GCSPOP)及認證碼檢查操作(authentication code checking operation, AUTIASP)。

```
myfunc():
PACIASP LR
GCSPUSH LR
... //我的功能內容

GCSPOP LR
```

AUTIASP LR

RET

【0093】 相比之下，藉由使用上文所描述之NOP相容指令提供兩段功能，此允許吾人具有較小的功能，且亦允許其在舊及新硬體上工作（因為舊硬體可視其為NOP，且甚至在較新硬體上，存在藉由設定指令功能選擇資訊至第一狀態而停用兩個操作之組態選項）。

【0094】 例如，若組合之指令（在此實例中，假定具有PACIASP編碼）執行所有操作，吾人可具有較小功能：

myfunc():
PACIASP LR //亦執行GCSPUSH

...//我的功能內容

AUTIASP LR //亦執行GCSPOP

RET

【0095】 應注意，AUTIASP係PACIASP之補體，且在上述實例中，藉由執行GCSPOP操作，亦為GCSPUSH之補體。

【0096】 舉例而言，吾等可提供2個控制位元以管理此等編碼之行為

00 - PAC停用，GCS停用

01 - PAC啟用，GCS停用

10 - PAC停用，GCS啟用

11 - PAC啟用，GCS啟用

【0097】 此可在未來延伸有新特徵，添加控制位元來開啟/關閉功能，全部由單一編碼執行。此外，控制狀態將可能控制操作被執行的相對順序。

【0098】 因此，多個新特徵可開啟及關閉相同指令編碼，而不需重建程式二元。

【0099】 上述實例展示第一操作及第二操作係以下任一者的情況：

- 針對功能導言變體，分別係 GCS 推送操作及認證碼產生操作，或反之亦然；或
- 針對功能收尾變體，分別係 GCS 取出操作及認證碼檢查操作，或反之亦然。

【0100】 兩種變體可在相同ISA中受支援，其中分別針對功能導言變體及功能收尾變體有運算碼72的不同編碼。

【0101】 然而，在其他實例中，亦可能提供NOP相容指令，其組合GCS推送操作與認證碼產生操作之外的另一種操作，或其組合認證碼產生操作與GCS推送操作以外的另一種操作。類似地，可能提供NOP相容指令，其組合GCS取出操作與認證碼檢查操作之外的另一種操作，或其組合認證碼檢查操作與GCS取出操作以外的另一種操作。

【0102】 圖9至圖12顯示在回應於NOP相容指令而執行兩操作之情況中，控制PAC/AUT及GCS操作之間的順序的不同實例。

【0103】 圖9顯示其中NOP相容指令（作為功能導言執行）執行認證碼產生操作(PAC)及GCS推送操作的實例，其中GCS推送操作取決於認證碼產生操作的結果，使得功能返回位址及其相關聯認證碼均被推送至

GCS。換言之，此等效於執行PAC操作且接著繼續執行GCS推送操作，其中，PAC操作之目的地暫存器相同於用於GCS推送操作之來源暫存器。相比之下，圖10顯示以不同順序執行PAC及GCS推送操作，其中GCS推送操作獨立於PAC操作。此允許功能返回位址至GCS結構之推送與PAC操作的認證碼的計算並行地開始。此可有用，因為PAC產生功能140及用於GCS推送操作之記憶體存取可相對緩慢，且因此相較於圖9之順序，使此等操作並行可具有改善之效能。另一方面，圖9的實例可改善安全性，因為所產生的認證碼在GCS上被保護，不僅是功能返回位址被保護。

【0104】 類似地，圖11及圖12顯示其中NOP相容指令（作為功能收尾執行）實施GCS取出操作及認證碼檢查(AUT)操作作為第一操作及第二操作之實例的替代順序（任一方式-第一操作可係GCS取出操作，且第二操作可係AUT操作，或反之亦然）。圖11展示一順序，其中AUT操作取決於GCS取出行操作之結果，因為由GCS取出操作從GCS取出的值係用作AUT操作的輸入。此方法可用於一實例中，其中執行作為功能導言之對應NOP相容指令使用圖9所示之方法。再次，此具有較大安全性的優點，因為使用由MMU 28實施的GCS記憶體區域保護來保護認證碼免受竄改。另一方面，圖12顯示一順序，其中AUT操作及GCS取出操作係獨立，使得其等可並行開始。在此情況下，認證碼檢查可施加至連結暫存器中的值，且接著分開地，GCS取出操作亦可取出至目的地暫存器的值（不具有關聯的認證碼）。假設在自GCS結構取出的值自記憶體被返回之前，針對AUT操作讀取連結暫存器中的舊值，GCS取出操作的目的地暫存器仍可係相同於用於AUT操作之源運算元的暫存器，在此情況下，若AUT操作偵測到失

配，採取錯誤或其他錯誤回應動作，且若程式碼匹配，則允許GCS取出操作完成，且接著，由GCS取出操作取出的位址可用於功能返回。替代地，GCS取出操作可將來自記憶體之GCS結構的經保護返回位址取出至由AUT操作所檢查之暫存器以外的暫存器，且接著隨後可進行由AUT操作所檢查的值與自GCS結構所取出之值之間的比較，以確認GCS保護位址是否匹配AUT檢查位址，從而提供進一步檢查是否基於該位址進行功能返回係安全。

【0105】 因此，有用的是指令解碼器10及執行級16支援：

- 一第一 NOP 相容指令用於一功能導言，其中該第一/第二操作中之一者係該認證碼產生操作(PAC)，且該第一/第二操作中之另一者係 GCS 推送操作；及
- 一第二 NOP 相容指令用於一功能收尾（具有與該第一 NOP 相容指令不同之編碼），其中該第一/第二操作中之一者係該認證碼檢查操作(AUT)，且該第一/第二操作中之另一者係 GCS 取出操作。

【0106】 指令功能選擇資訊60可指定第一操作及第二操作（若有）的何者將回應於該指令執行，且亦可控制如圖9至圖12所示之操作的相對順序。

【0107】 圖13係顯示藉由MMU 28執行之步驟的流程圖，用於檢查記憶體存取請求的記憶體存取權限。在步驟200，觸發載入/儲存操作的指令被解碼。若指令功能選擇資訊60指定待執行GCS推送或取出操作，則支援GCS推送或取出操作的上述NOP相容指令被視為一載入/儲存指令。

【0108】 在步驟202，判定觸發載入/儲存操作的指令是否是GCS存取指令（被允許存取記憶體的GCS區域之GCS存取指令之限制子集中之一者）。當指令功能選擇資訊指示需要GCS推送或取出操作時，將NOP相容指令視為GCS存取指令。亦可將其他類型的指令視為GCS存取指令。在一些情況下，若指令功能選擇資訊在指示不需要GCS推送或取出操作的狀態中，不將NOP相容指令視為GCS存取指令。替代地，若GCS推送或取出操作係能夠回應於NOP相容指令產生載入/儲存操作的唯一操作（例如，上文描述之PAC/AUT操作無法產生任何載入/儲存請求），則NOP相容指令可始終被視為GCS存取指令，無論該指令功能選擇資訊60的值如何。

【0109】 若經解碼指令係GCS存取指令，則在步驟204處，MMU 28檢查對應於載入/儲存操作所存取的目標位址之記憶體屬性資料是否指定對應於目標位址的記憶體位址空間區域係GCS區域。此記憶體屬性資料可衍生自對應於目標位址的頁表項目或由頁表項目指定的間接暫存器（且可在MMU 28之轉譯後備緩衝區(TLB)中被快取）。若指令係GCS存取指令，但記憶體屬性資料指定對應於目標位址之區域不為GCS區域，則在步驟206處，載入/儲存操作被拒絕，且傳訊錯誤。此防止GCS存取指令用於存取非GCS記憶體，此更安全，以減少可用於攻擊器之攻擊面。否則，若GCS存取指令存取GCS區域，則在步驟212處，MMU 28檢查任何其他存取權限檢查是否由載入/儲存操作傳遞。若此等其他檢查失敗，則在步驟214處，載入/儲存操作被拒絕，且傳訊錯誤。若傳遞此等其他檢查，則在步驟216處，允許載入/儲存操作。此等其他檢查可例如檢查指示是否允許讀取或寫入至記憶體區域之讀取/寫入權限，或可檢查不相關於GCS區域

檢查的其他安全屬性（例如，限制允許存取記憶體區域中的處理器的執行狀態之屬性）。

【0110】 若觸發載入/儲存操作的指令並非GCS存取指令，則在步驟208處，MMU 28檢查對應於載入/儲存操作所存取的目標位址之記憶體屬性資料是否指定對應於目標位址的記憶體位址空間區域係GCS區域。然而，在此情況下，相較於步驟204，該回應相反，若非GCS存取指令不存取GCS區域，載入/儲存請求可可能被接受，而若其存取GCS區域，可被拒絕。更具體而言，若對應於目標位址之區域的記憶體屬性資料指定GCS區域，則在步驟205處，MMU檢查記憶體存取是否係寫入記憶體存取，且若是，在步驟210處拒絕載入/儲存操作，且傳訊錯誤。藉由限制至GCS區域的寫入存取至GCS存取指令的專用類別，此防止程式碼中的大部分正規儲存指令竄改GCS資料結構上的保護的返回狀態。此減少攻擊器在嘗試損壞GCS上之保護的返回狀態時修改正規儲存指令的運算元之機會。

【0111】 另一方面，若非GCS存取指令不嘗試存取GCS區域（在步驟208處，否）或嘗試存取GCS區域，但係讀取請求（在步驟205處，否），則該方法再次繼續進行至步驟212，以便施加任何其他存取權限檢查，且接著取決於此等檢查的結果而控制載入/儲存操作是否被拒絕或允許，相同如上文針對步驟212、214、216所論述的。

【0112】 圖14繪示可使用的模擬器實施方案。雖然稍早所述之實施例以用於操作支援所關注技術的特定處理硬體之設備及方法來實施本發明，但亦可能根據本文所述之實施例提供一指令執行環境，其係透過使用電腦程式實施。此類電腦程式常稱為模擬器，因為其等提供硬體架構之基

於軟體的實施方案。模擬器電腦程式的種類包括仿真器、虛擬機、模型、及二元轉譯器（包括動態二元轉譯器）。一般而言，模擬器實施方案可在可選地運行主機作業系統1320、支援模擬器程式1310的主機處理器1330上運行。在一些配置中，在硬體與所提供的指令執行環境及/或相同的主機處理器上提供的多個相異指令執行環境之間可有多層模擬。歷史上，已需要強大的處理器來提供模擬器實施方案，其以合理速度執行，但此種方法在某些情況下可係有正當理由的，諸如當因為相容性或再使用原因此需要執行另一處理器原生的程式碼時。例如，模擬器實施方案可提供具有不為主機處理器硬體所支援之額外功能性的指令執行環境，或提供一般與不同的硬體架構相關聯的指令執行環境。模擬的綜述係於「Some Efficient Architecture Simulation Techniques」中給出，Robert Bedichek, Winter 1990 USENIX Conference，頁數53至63。

【0113】 在先前已參照特定硬體架構或特徵來描述實施例之情況下，在一模擬實施例中，可藉由合適的軟體架構或特徵提供等效功能。例如，可在模擬實施例中將特定電路系統實施為電腦程式邏輯。類似地，記憶體硬體（諸如暫存器或快取）可在模擬實施例中實施為軟體資料結構。在先前描述的實施例中提及的硬體元件的一或多者存在於主機硬體（例如，主機處理器1330）上的配置中，一些模擬實施例可（在適當處）利用主機硬體。

【0114】 模擬器程式1310可儲存在電腦可讀儲存媒體（其可係非暫時性媒體）上，並提供程式介面（指令執行環境）給目標碼1300（其可包括應用程式、作業系統、及超管理器），該程式介面與藉由模擬器程式

1310模型化之硬體架構的介面相同。因此，目標碼1300之包括如上文描述之NOP相容指令的程式指令可在指令執行環境內使用模擬器程式1310執行，使得實際上不具有上文討論之設備2的硬體特徵的主機電腦1330可模仿此等特徵。類似地，圖13的記憶體管理檢查功能可使用模擬器程式1310的記憶體管理程式邏輯1318來仿真。

【0115】 因此，模擬器程式1310可具有處理程式邏輯1312，其模擬用於硬體設備2之上述處理的狀態。例如，處理程式邏輯1312可回應於目標碼1300之模擬執行期間發生的事件而控制執行狀態（例如，例外層級）的轉變。指令解碼程式邏輯1314仿真指令解碼器10之行為，且解碼目標碼1300之指令，且將此等指令映射至主機設備1330之原生指令集中的對應指令集。暫存器仿真程式邏輯1316將由目標碼請求的暫存器存取映射成對維護在主機設備1330之主機硬體上的對應資料結構的存取，諸如藉由存取主機設備1330之暫存器或記憶體1332中的資料。記憶體管理程式邏輯1318實施位址轉譯、頁表走訪及以對應於在上文的硬體實施例中描述的MMU 28之方式存取控制檢查，但亦具有映射模擬實體位址（基於針對目標碼1300所界定的頁表而藉由位址轉譯獲得）的額外功能至用以存取主機記憶體1332的主機虛擬位址。此等主機虛擬位址可本身使用由主機所支援之標準位址轉譯機構來轉譯成主機實體位址（將主機虛擬位址轉譯成主機實體位址係在受到模擬器程式1310控制的範圍之外）。

【0116】 在本申請案中，用語「經組態以...(configured to...)」係用以意指一設備的一元件具有能夠實行該經定義作業的一組態。在此上下文中，「組態(configuration)」意指硬體或軟體之互連的配置或方式。例如，

該設備可具有專用硬體，其提供經定義的作業，或者一處理器或其他處理裝置可經程式化以執行該功能。「經組態以(configured to)」並不意味著設備元件需要以任何方式改變以提供所定義的作業。

【0117】 雖然本文已參照附圖詳細地描述本發明的說明性實施例，應瞭解本發明不限於該等精確實施例，且所屬技術領域中具有通常知識者可於其中實行各種變化與修改，而不脫離如隨附申請專利範圍所定義的本發明的範圍。

【符號說明】

【0118】

2:資料處理設備

4:處理管線

6:提取級

8:指令快取記憶體/記憶體系統/快取記憶體

10:解碼級

12:發布級

14:暫存器檔案

16:執行級

18:寫回級

20:算術/邏輯單元(arithmetic/logic unit, ALU)/處理單元/執行單元

22:浮點單元/處理單元/執行單元

24:分支單元/處理單元/執行單元

26:載入/儲存單元/處理單元/執行單元

28:記憶體管理單元(memory management unit, MMU)

29:轉譯後備緩衝區(translation lookaside buffer, TLB)

30:記憶體系統/快取記憶體

32:記憶體系統/快取記憶體

34:記憶體系統

50:通用暫存器

52:堆疊指標(stack pointer, SP)暫存器

54:連結暫存器

56:控制暫存器

58:保護控制堆疊(GCS)堆疊指標暫存器/GCS 指標暫存器

60:指令功能選擇資訊

70:NOP 相容指令

72:運算碼

74,76:運算元欄位/運算元

120:資料結構/GCS 結構/GCS 資料結構

100,102,104,106,108,110,112,114,140:步驟

140:認證碼產生功能

200,202,204,205,206,208,210,212,214,216:步驟

1300:目標碼

1310:模擬器程式

1312:處理程式邏輯

1314:指令解碼程式邏輯

1316:暫存器仿真程式邏輯

1318:記憶體管理程式邏輯

1320:主機作業系統

1330:主機處理器/主機設備/主機電腦

1332:暫存器或記憶體/主機記憶體

【發明申請專利範圍】

【請求項1】一種設備，其包含：

一指令解碼器，以解碼指令；

處理電路系統，以回應於該等指令藉由該指令解碼器的解碼而執行資料處理；及

至少一控制暫存器，以指定指令功能選擇資訊；

其中：

回應於一無操作相容指令，該指令解碼器經組態以控制該處理電路系統以：

當由該至少一控制暫存器指定之該指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；

當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及

當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【請求項2】如請求項1之設備，其中當該指令功能選擇資訊在一第四狀態中時，回應於該無操作相容指令，該指令解碼器經組態以控制該處理電路系統，以執行該第二操作，但不執行該第一操作。

【請求項3】如前述請求項中任一項之設備，其中當該指令功能選擇資訊係在該第二狀態中時，該處理電路系統經組態以基於該指令功能選擇資訊控制施加該第一操作與該第二操作的一相對順序。

【請求項4】如前述請求項中任一項之設備，其中該第一操作及該第二操作係與一功能呼叫相關聯的功能導言操作；或

該第一操作及該第二操作係與來自處理一功能之一返回相關聯的功能收尾操作。

【請求項5】如前述請求項中任一項之設備，其中對於該無操作相容指令之至少一變體，該第一操作及該第二操作中之一者包含一認證碼產生操作，以基於一運算元而產生一認證碼，且將該認證碼與該運算元相關聯。

【請求項6】如請求項5之設備，其中該運算元包含自一連結暫存器獲得之一值；及

回應於一功能返回分支指令，該指令解碼器經組態以控制該處理電路系統以分支至該連結暫存器中指定的一位址。

【請求項7】如請求項5及6中任一項之設備，其中將該認證碼與該運算元相關聯包含將該認證碼嵌入在該運算元之更高有效位元的一部分中。

【請求項8】如請求項5至7中任一項之設備，其中該認證碼產生操作包含至少基於該運算元及一密鑰根據一密碼功能而產生該認證碼。

【請求項9】如前述請求項中任一項之設備，其中對於該無操作相容指令之至少一變體，該第一操作及該第二操作中之一者包含一保護控制堆疊(guarded-control-stack, GCS)推送操作，以將該運算元推送至用於保護返回狀態資訊的一GCS資料結構。

【請求項10】如前述請求項中任一項之設備，其中：

對於該無操作相容指令之至少一變體：

該第一操作及該第二操作中之一者包含一認證碼產生操作，以基於一運算元而產生一認證碼，且將該認證碼與該運算元相關聯；

該第一操作及該第二操作中之另一者包含一保護控制堆疊(GCS)推送操作，以將該運算元推送至用於保護返回狀態資訊的一GCS資料結構。

【請求項11】如請求項10之設備，其中：

回應於該無操作相容指令，當該指令功能選擇資訊係在該第二狀態之一第一子狀態中時，該處理電路系統經組態以將該運算元及該認證碼推送至該GCS資料結構；及

回應於該無操作相容指令，當該指令功能選擇資訊係在該第二狀態之一第二子狀態中時，該處理電路系統經組態以推送該運算元、但不推送該認證碼至該GCS資料結構。

【請求項12】如前述請求項中任一項之設備，其中對於該無操作相容指令之至少一變體：

該第一操作及該第二操作中之一者包含一認證碼檢查操作，以檢查與一運算元相關聯的一相關聯認證碼是否匹配於基於該運算元產生的一預期認證碼，且回應於偵測到該相關聯認證碼及該預期認證碼之間的一失配而觸發一錯誤處理回應。

【請求項13】如請求項12之設備，其中該相關聯認證碼係自該運算元之更高有效位元的一部分獲得。

【請求項14】如前述請求項中任一項之設備，其中對於該無操作相容指令之至少一變體，該第一操作及該第二操作中之一者包含一保護控制堆疊(GCS)取出操作，以自用於保護該功能返回資訊的一GCS資料結構取出功能返回資訊。

【請求項15】如前述請求項中任一項之設備，其中對於該無操作相容指令之至少一變體：

該第一操作及該第二操作中之一者包含一認證碼檢查操作，以檢查與一運算元相關聯的一相關聯認證碼是否匹配於基於該運算元產生的一預期認證碼，且回應於偵測到該相關聯認證碼及該預期認證碼之間的一失配而觸發一錯誤處理回應；及

該第一操作及該第二操作中之另一者包含一保護控制堆疊(GCS)取出操作，以自用於保護該功能返回資訊的一GCS資料結構取出功能返回資訊。

【請求項16】如請求項15之設備，其中：

回應於該無操作相容指令，當該指令功能選擇資訊係在該第二狀態之一第一子狀態中時，該處理電路系統經組態以執行該GCS取出操作，並執行該認證碼檢查操作於由該GCS取出操作自該GCS資料結構取出的一值；及

回應於該無操作相容指令，當該指令功能選擇資訊係在該第二狀態之一第二子狀態中時，該處理電路系統經組態以在執行該GCS取出操作之前執行該認證碼檢查操作於一給定暫存器中的一值，並執行該GCS取出操作，以自該GCS資料結構取出該功能返回資訊至該給定暫存器。

【請求項17】如請求項9、10、11、14、15及16中任一項之設備，其中：

回應於該無操作相容指令，當該指令功能選擇資訊係在指示對該GCS資料結構的一存取將回應於該無操作相容指令被執行之一狀態中時：

該處理電路系統經組態以回應於偵測到對應於該無操作相容指令之一目標位址的一記憶體區域係由記憶體屬性資料指定為用於儲存該GCS資料結構之一

GCS區域以外的一記憶體區域，而拒絕該無操作相容指令所觸發的一記憶體存取。

【請求項18】如請求項17之設備，其中該處理電路系統經組態以回應於偵測到對應於一非GCS存取類型指令之一目標位址的一記憶體區域係由記憶體屬性資料指定為該GCS區域，而拒絕該非GCS存取類型指令所觸發的一寫入記憶體存取。

【請求項19】如前述請求項中任一項之設備，其中該指令功能選擇資訊包含：一第一操作指示符，以指示該第一操作是否將回應於該無操作相容指令而執行；及一第二操作指示符，以指示該第二操作是否將回應於該無操作相容指令而執行。

【請求項20】如前述請求項中任一項之設備，其中該指令功能選擇資訊進一步指示該處理電路系統是否應回應於該無操作相容指令而執行一第三操作。

【請求項21】一種方法，其包含：

解碼指令；及

回應於該等指令之解碼而執行資料處理；其中：

回應於一無操作相容指令之解碼，該方法包含：

當由至少一控制暫存器指定之指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；

當由該至少一控制暫存器指定之該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及

當由該至少一控制暫存器指定之該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【請求項22】一種包含指令的電腦程式，當該等指令由一主機資料處理設備執行時，控制該主機資料處理設備提供用於執行目標碼的一指令執行環境，該電腦程式包含：

指令解碼程式邏輯，以解碼該目標碼之指令；及

暫存器仿真程式邏輯，以將資料維持在該主機資料處理設備的儲存電路系統中，以仿真至少一控制暫存器用於指定指令功能選擇資訊；其中：

回應於一無操作相容指令，該指令解碼程式邏輯經組態以控制該主機資料處理設備以：

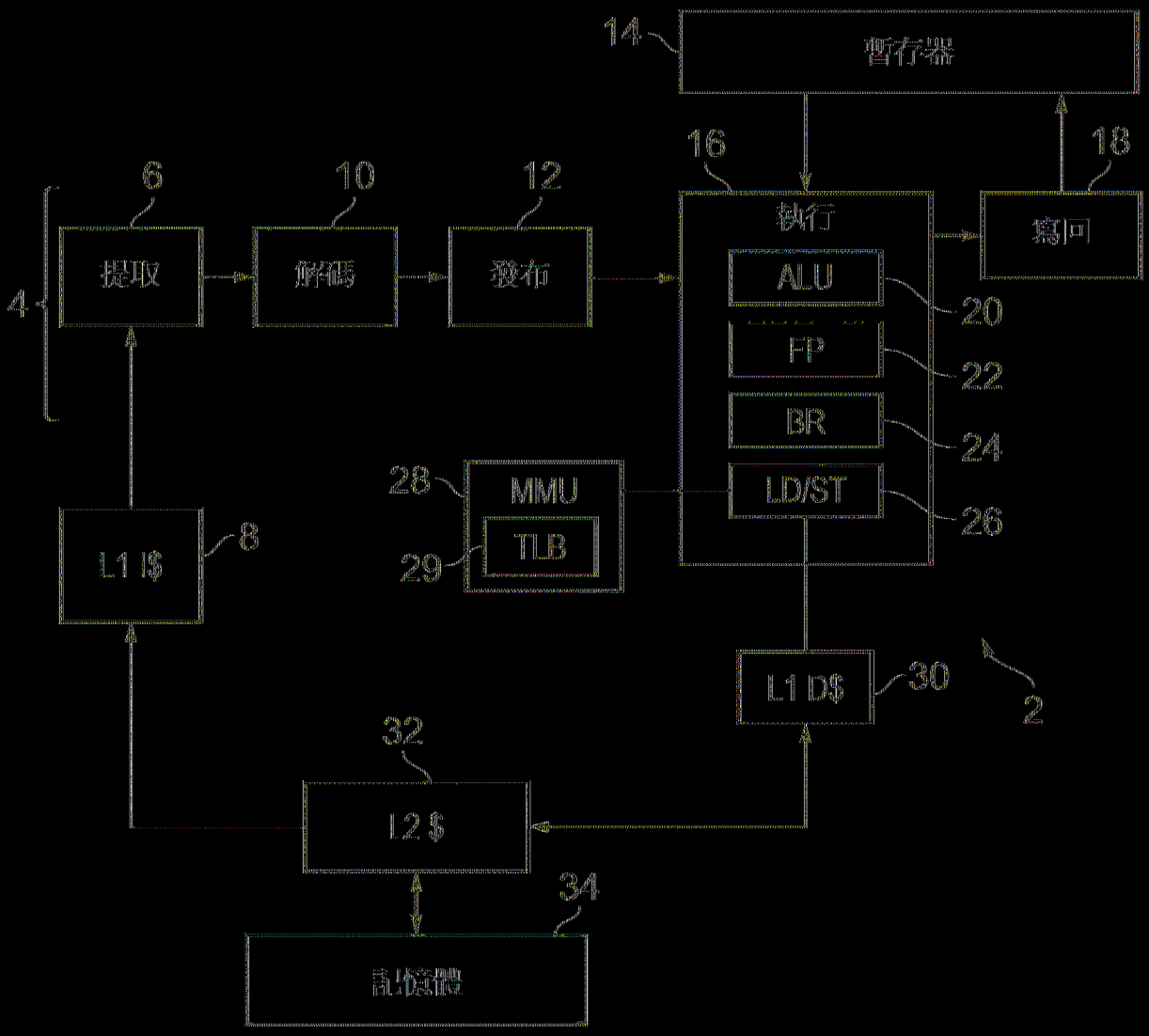
當該指令功能選擇資訊在一第一狀態中時，將該無操作相容指令視為一無操作指令；

當該指令功能選擇資訊在一第二狀態中時，執行一第一操作及一第二操作兩者；及

當該指令功能選擇資訊在一第三狀態中時，執行該第一操作但不執行該第二操作。

【請求項23】一種儲存媒體，其儲存如請求項22之電腦程式。

|(發明圖式)|


14
暫存器
16
18
6
10
12
執行
寫回
提取
解碼
發布
4
ALU
20
FP
22
BR
24
LD/ST
26
28
MMU
TLB
29
L1 I$
8
L1 D$
30
2
32
L2$
34
記憶體

|(圖1)|

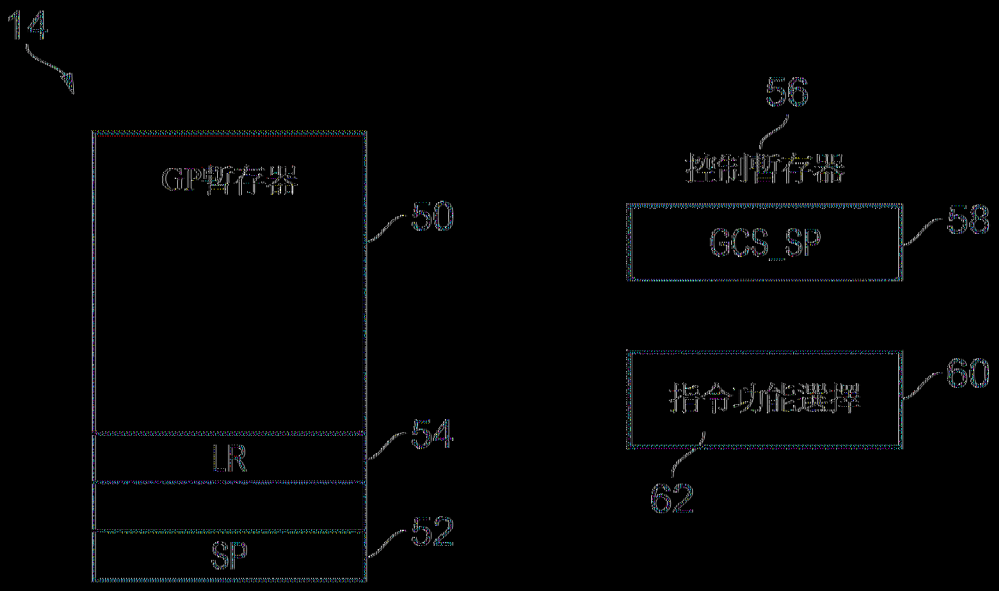

14
GP暫存器
50
LR
54
SP
52
56
控制暫存器
GCS_SP
58
指令功能選擇
60
62

圖2

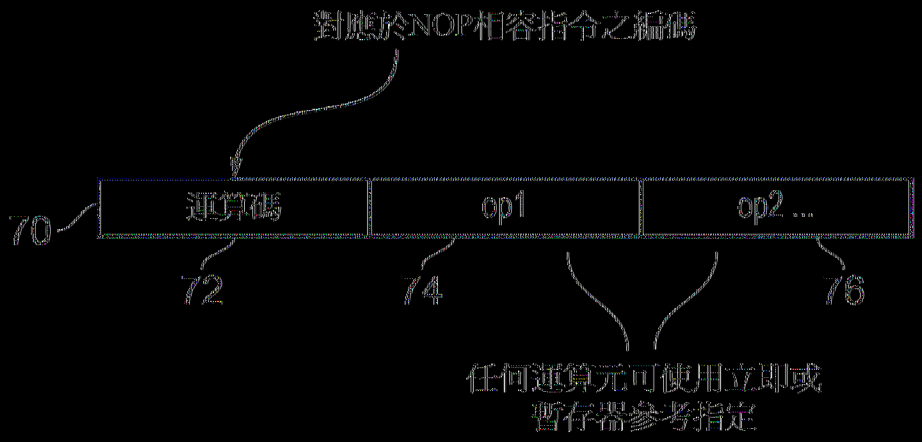

對應於NOP相容指令之編碼
運算碼
op1
op2 ...
70
72
74
76
任何運算元可使用立即或
暫存器參考指定

圖3

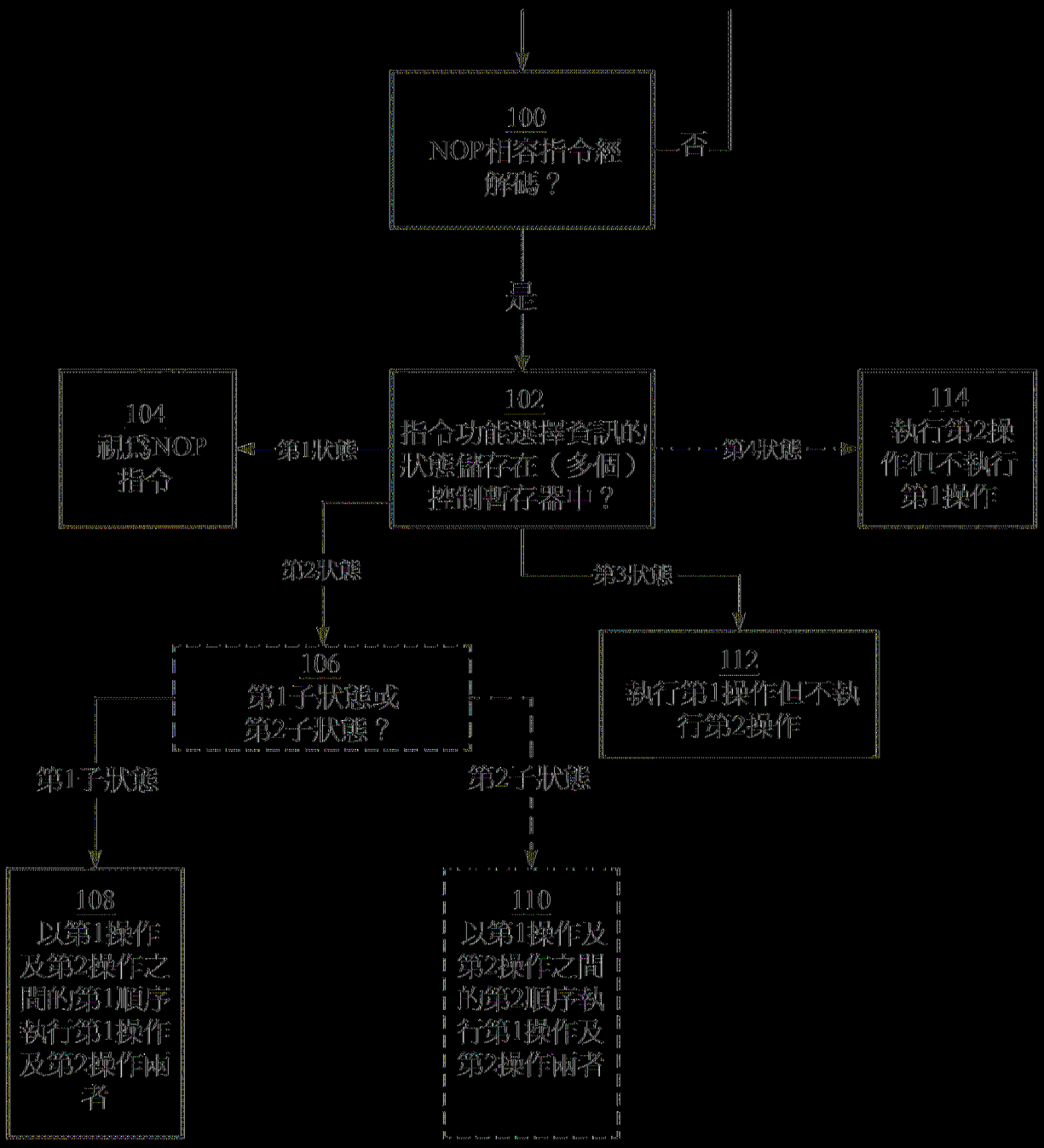

100
NOP相容指令經解碼？
否
是
102
指令功能選擇資訊的狀態儲存在（多個）控制暫存器中？
第1狀態
104
視為NOP指令
第4狀態
114
執行第2操作但不執行第1操作
第2狀態
第3狀態
106
第1子狀態或第2子狀態？
112
執行第1操作但不執行第2操作
第1子狀態
第2子狀態
108
以第1操作及第2操作之間的第1順序執行第1操作及第2操作兩者
110
以第1操作及第2操作之間的第2順序執行第1操作及第2操作兩者

圖4

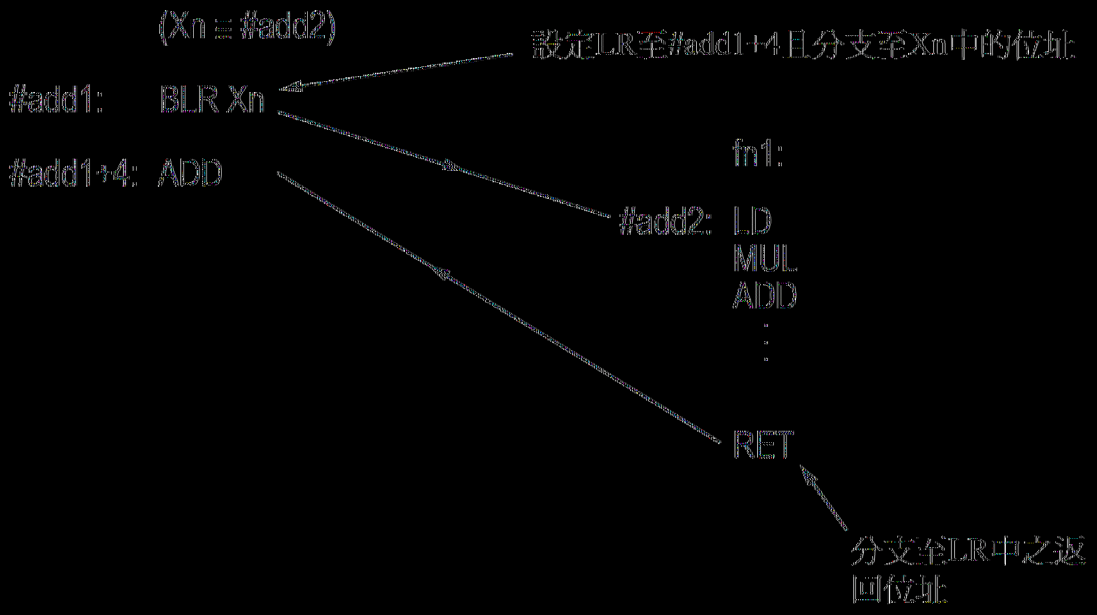

(Xn = #add2)
設定LR至#add1+4且分支至Xn中的位址
#add1:
BLR Xn
#add1+4:
ADD
fn1:
#add2:
LD
MUL
ADD
RET
分支至LR中之返回位址

圖5

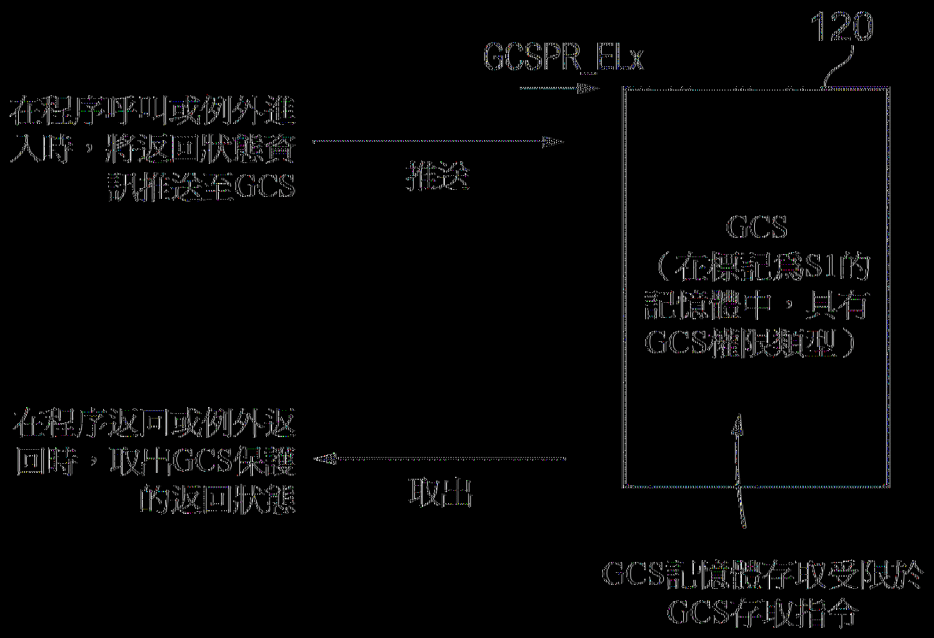
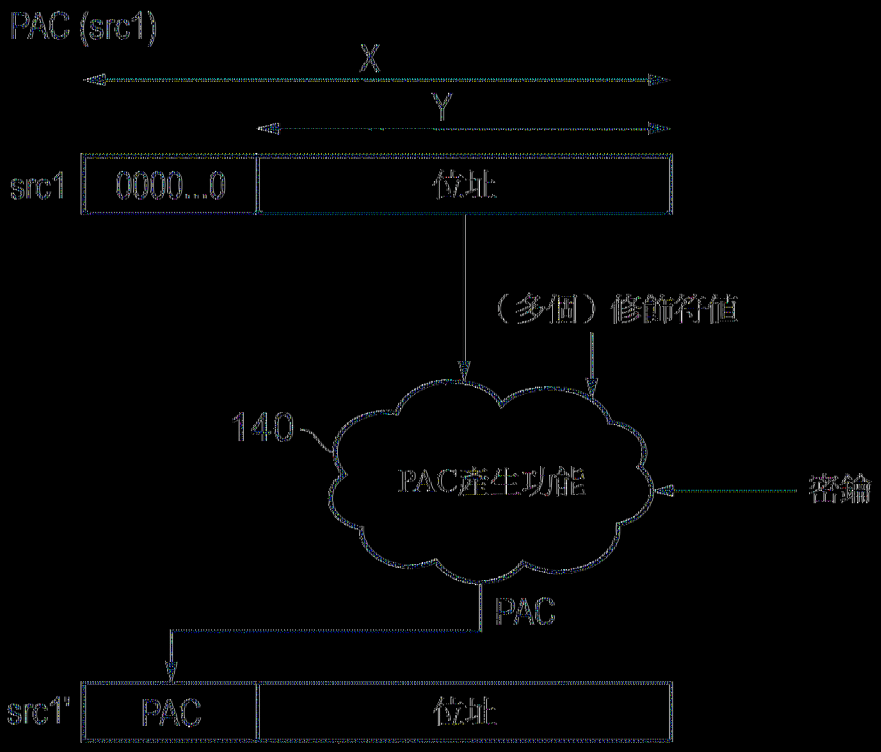
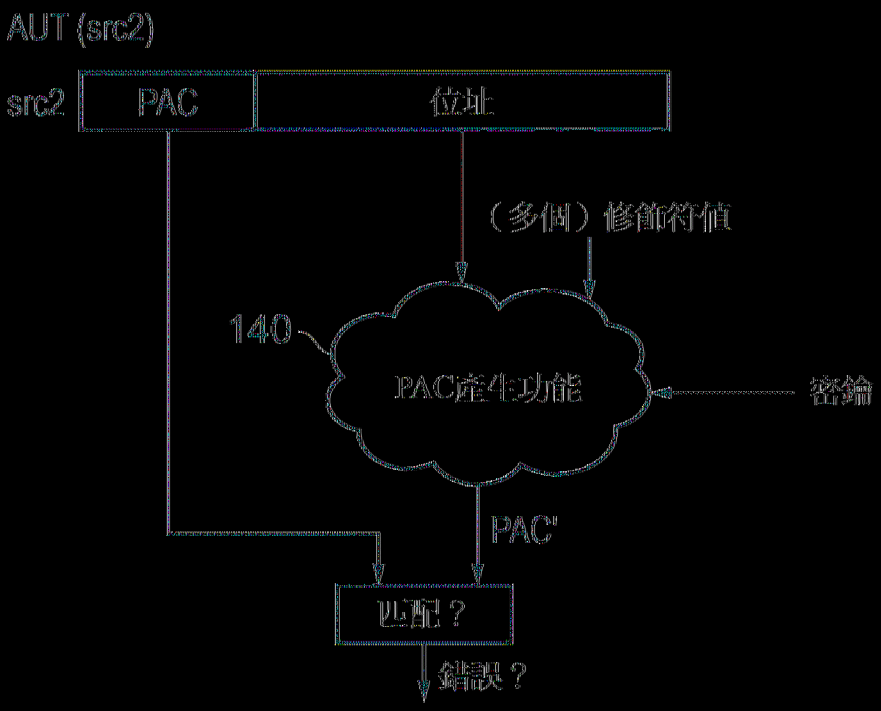

120
GCSPR_ELx
在程序呼叫或例外進入時，將返回狀態資訊推送至GCS
推送
GCS
（在標記為S1的記憶體中，具有GCS權限類型）
在程序返回或例外返回時，取出GCS保護的返回狀態
取出
GCS記憶體存取受限於GCS存取指令

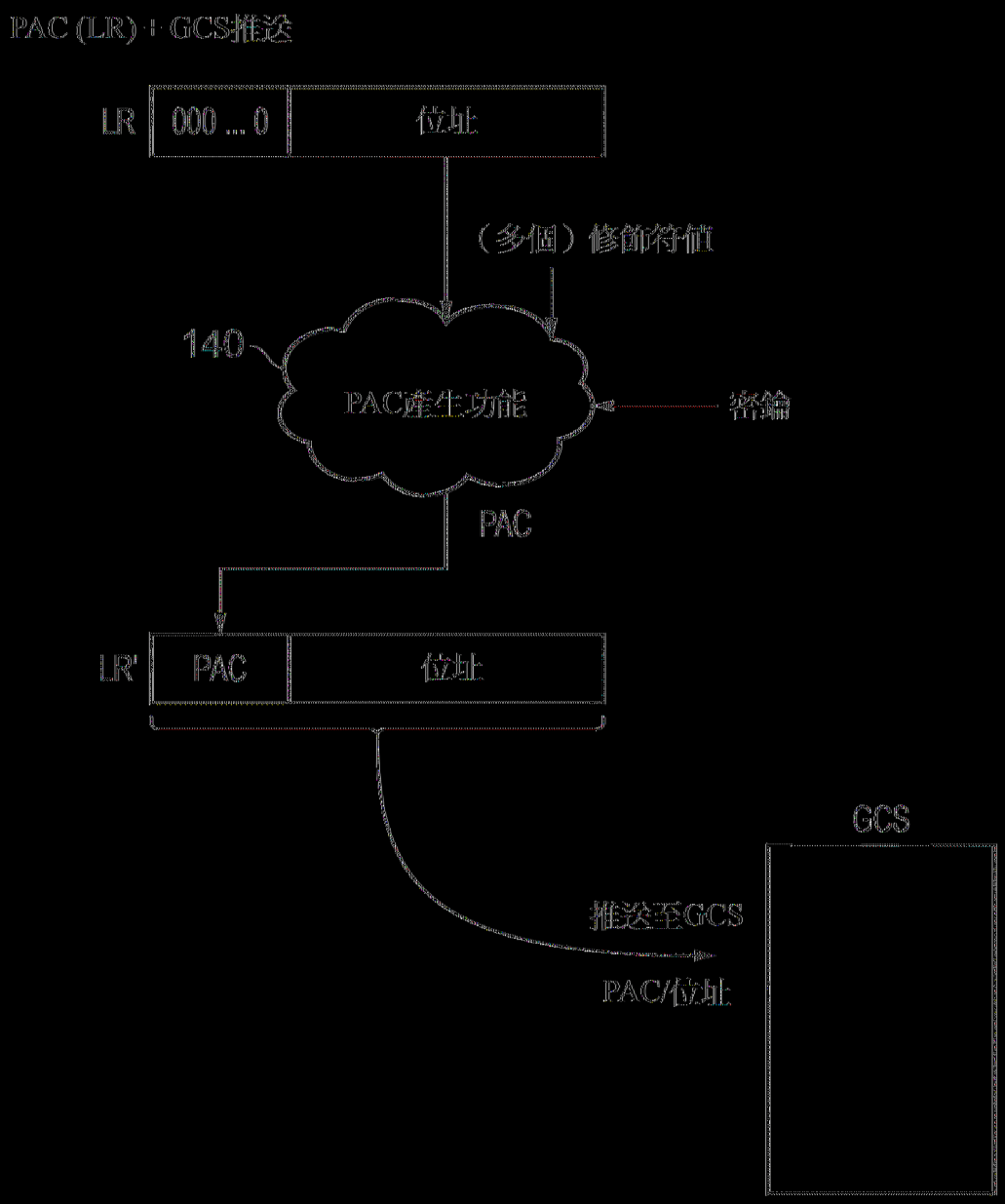
圖6


PAC (src1)
X
Y
src1
0000...0
位址
（多個）修飾符值
140
PAC產生功能
密鑰
PAC
src1'
PAC
位址

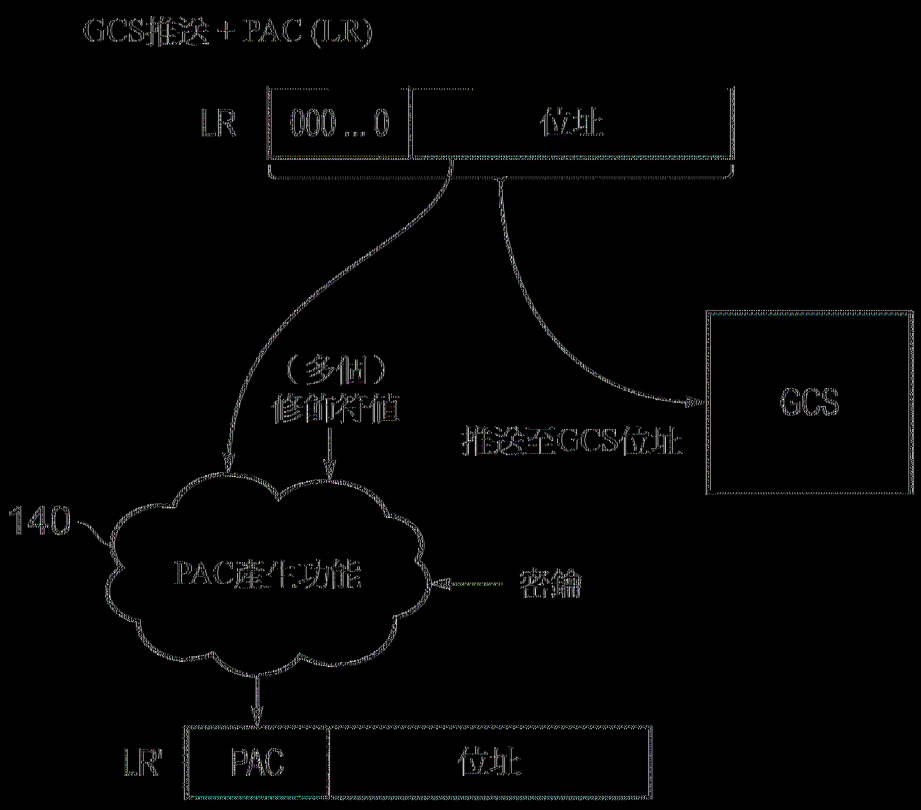
|(圖7)|


AUT (src2)
src2
PAC
位址
（多個）修飾符值
140
PAC產生功能
密鑰
PAC'
匹配？
錯誤？

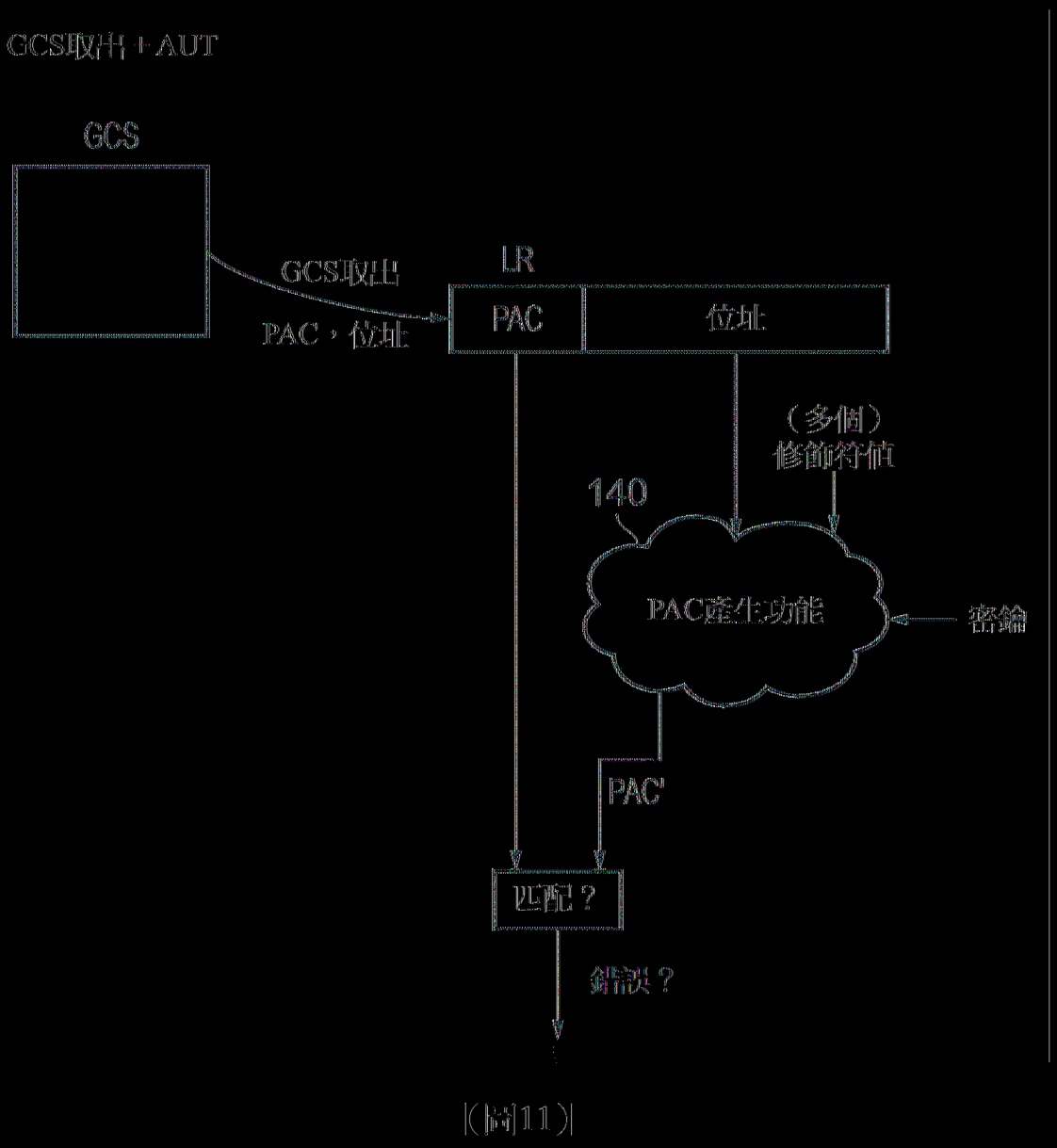
|(圖8)|

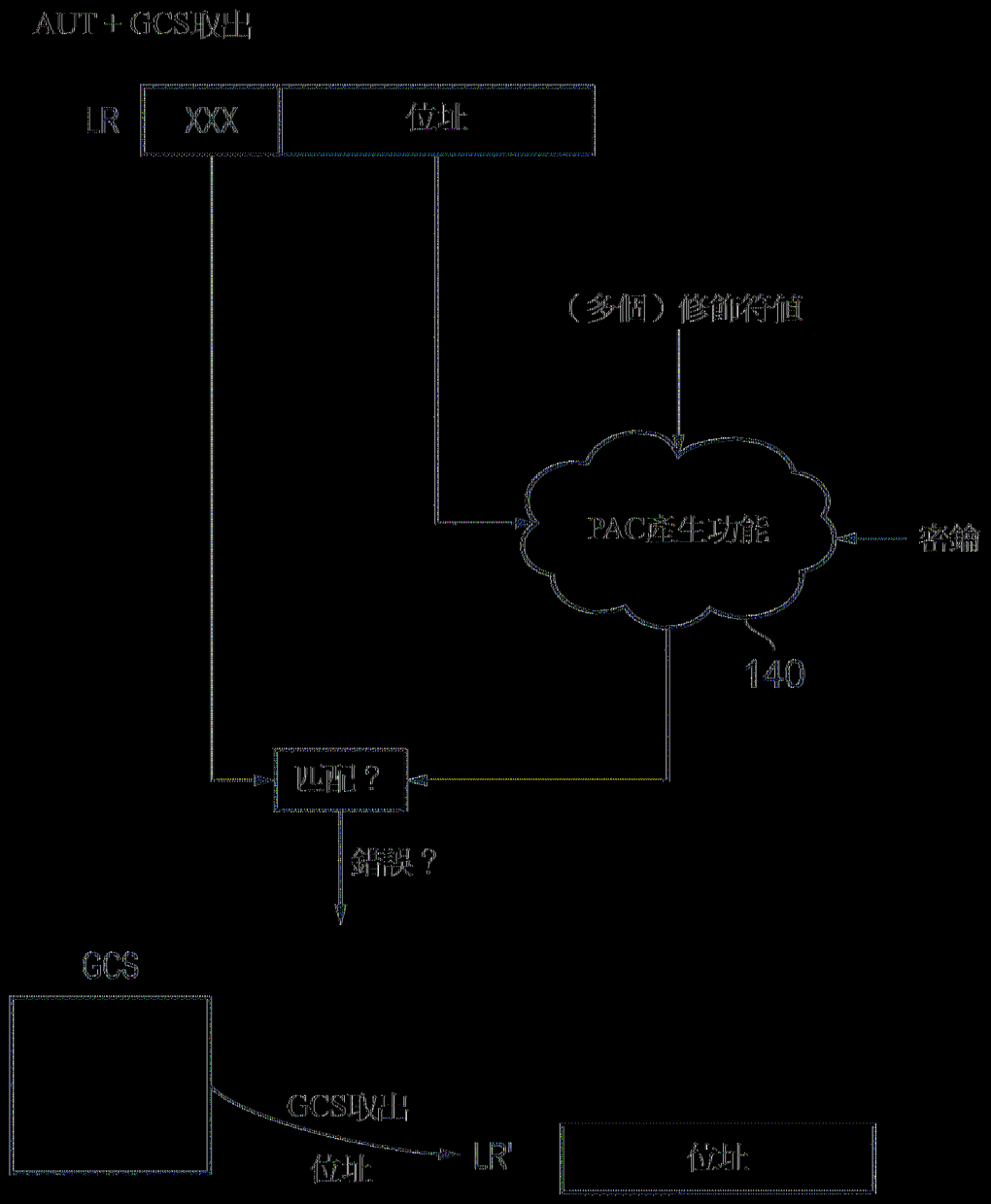

PAC (LR) + GCS推送
LR
000 ... 0
位址
（多個）修飾符值
140
PAC產生功能
密鑰
PAC
LR'
PAC
位址
GCS
推送至GCS
PAC/位址

|(圖9)|

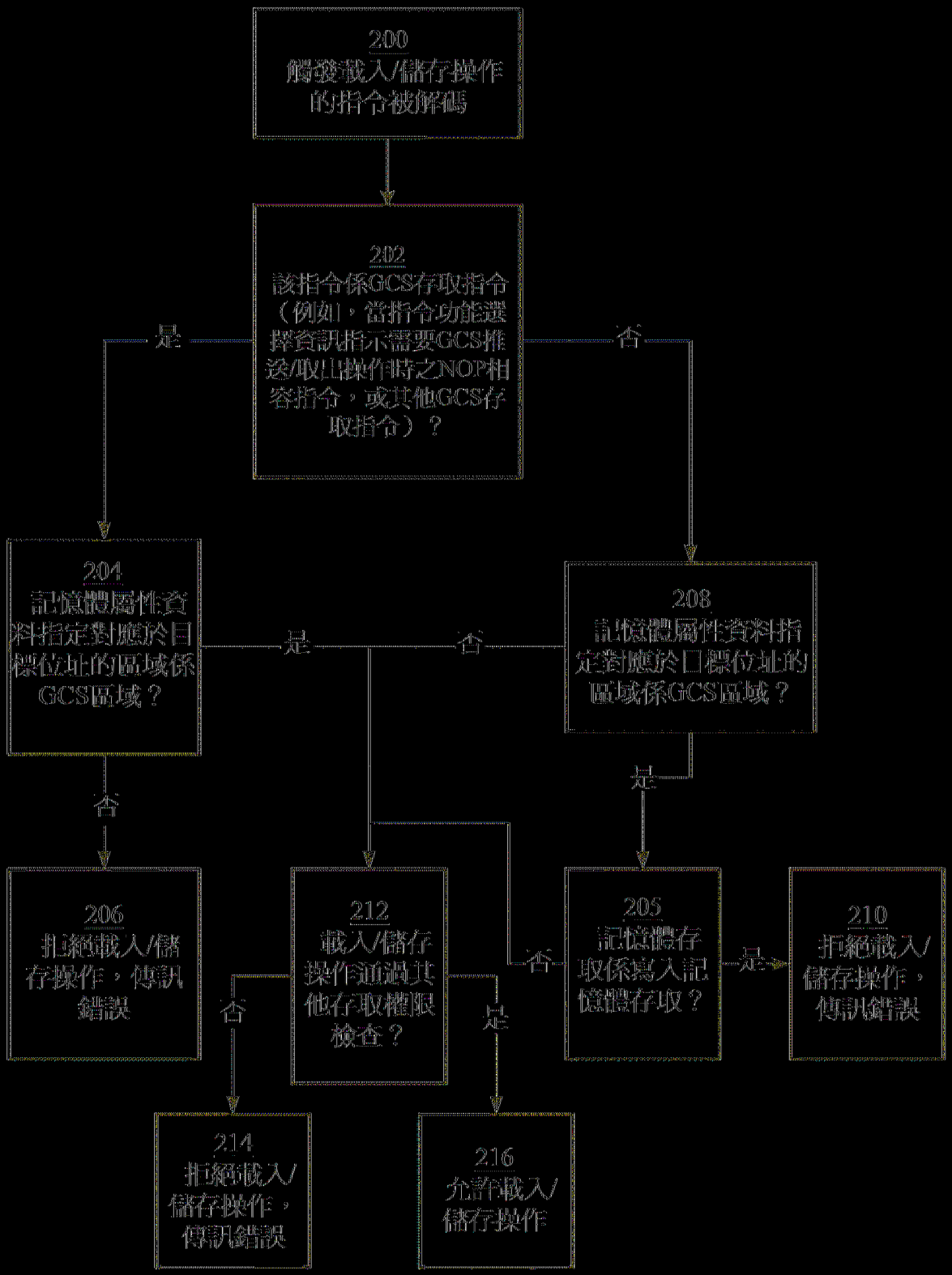

GCS推送 + PAC (LR)
LR
000...0
位址
(多個)
修飾符值
推送至GCS位址
GCS
140
PAC產生功能
密鑰
LR'
PAC
位址

|(圖10)|


GCS取出＋AUT
GCS
GCS取出
PAC，位址
LR
PAC
位址
（多個）
修飾符值
140
PAC產生功能
密鑰
PAC'
匹配？
錯誤？
（同11）


AUT + GCS取出
LR
XXX
位址
（多個）修飾符值
PAC產生功能
答錄
140
匹配？
錯誤？
GCS
GCS取出
位址
LR'
位址

[圖12]


200
觸發載入/儲存操作的指令被解碼
202
該指令係GCS存取指令（例如，當指令功能選擇資訊指示需要GCS推送/取出操作時之NOP相容指令，或其他GCS存取指令）？
是
否
204
記憶體屬性資料指定對應於目標位址的區域係GCS區域？
208
記憶體屬性資料指定對應於目標位址的區域係GCS區域？
是
否
否
是
206
拒絕載入/儲存操作，傳訊錯誤
212
載入/儲存操作通過其他存取權限檢查？
205
記憶體存取係寫入記憶體存取？
否
是
210
拒絕載入/儲存操作，傳訊錯誤
否
是
214
拒絕載入/儲存操作，傳訊錯誤
216
允許載入/儲存操作

[圖13]

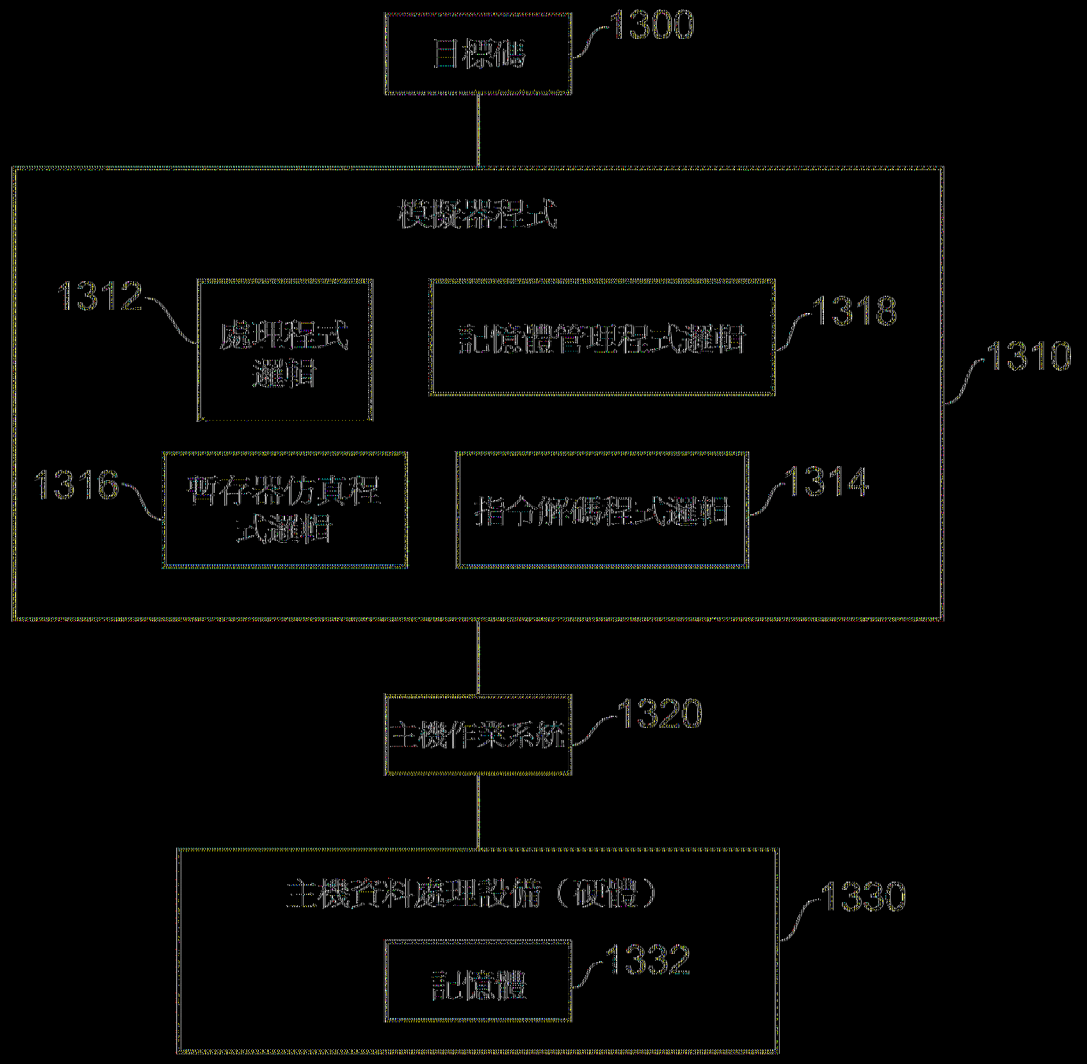

目標碼
1300
模擬器程式
1312
處理程式邏輯
記憶體管理程式邏輯
1318
1310
1316
暫存器仿真程式邏輯
指令解碼程式邏輯
1314
主機作業系統
1320
主機資料處理設備（硬體）
1330
記憶體
1332

圖14