



(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 201439772 A

(43)公開日：中華民國 103 (2014) 年 10 月 16 日

(21)申請案號：103101884

(22)申請日：中華民國 103 (2014) 年 01 月 17 日

(51)Int. Cl.：

G06F13/16 (2006.01)

G06F13/36 (2006.01)

G06F5/06 (2006.01)

(30)優先權：2013/02/04

美國

13/758,853

(71)申請人：L S I 公司 (美國) LSI CORPORATION

(US)

美國

(72)發明人：伯特 盧卡 BERT, LUCA (US)；席蒙內司克 霍利亞 SIMIONESCU, HORIA

(US)；巴德帝倪 雅南特 BADERDINNI, ANANT (IN)；伊許 馬克 ISH, MARK

(US)

(74)代理人：陳長文

申請實體審查：無 申請專利範圍項數：19 項 圖式數：5 共 30 頁

(54)名稱

用於藉由使用一命令推動模型減少一資料儲存系統中之寫入延遲之方法及系統

METHOD AND SYSTEM FOR REDUCING WRITE LATENCY IN A DATA STORAGE SYSTEM BY USING A COMMAND-PUSH MODEL

(57)摘要

本發明揭示一種資料儲存系統，其實施減少延遲之一命令推動模型。主機系統存取記憶體控制器之一非揮發性記憶體(NVM)裝置，以容許該主機系統將命令推動至定位於該 NVM 裝置中之一命令佇列中。該主機系統在不需要該記憶體控制器介入之情況下完成各 IO，從而消除該主機系統與該記憶體控制器之間之同步或交握之需要。對於寫入命令，記憶體控制器並不需要在完成該命令之後發佈一完成中斷至該主機系統，此係因為該主機系統在該寫入命令被推動至該記憶體控制器之該佇列中時認為該寫入命令已完成。全部此等特徵之組合導致總延遲之大量減少。

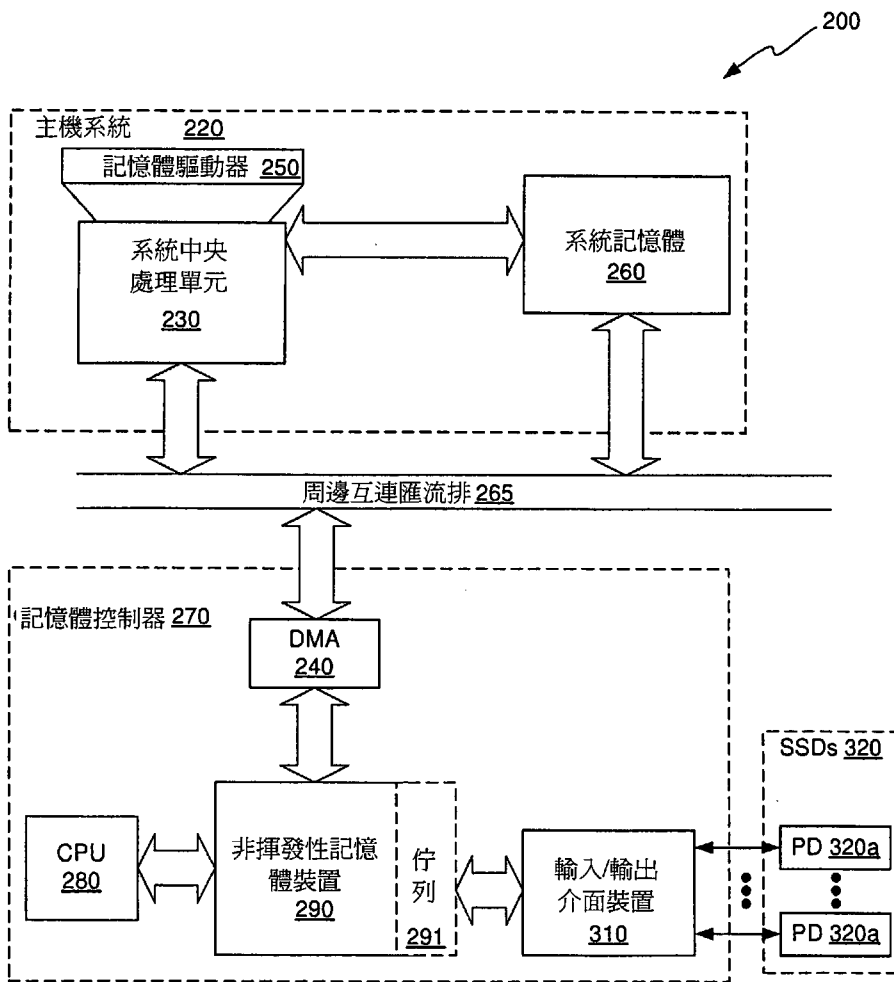


圖3

- 200：資料儲存系統/系統
- 220：主機系統
- 230：系統中央處理單元(CPU)/中央處理單元(CPU)
- 240：直接記憶體存取(DMA)引擎
- 250：記憶體驅動器
- 260：系統記憶體裝置
- 265：周邊互連(PCI)/周邊互連高速(PCIe)匯流排
- 270：記憶體控制器
- 280：控制器中央處理單元(CPU)/中央處理單元(CPU)
- 290：非揮發性記憶體(NVM)裝置
- 291：命令佇列/佇列
- 310：輸入/輸出(I/O)介面裝置
- 320：固態驅動機(SSD)
- 320a：實體磁碟(PD)



(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 201439772 A

(43)公開日：中華民國 103 (2014) 年 10 月 16 日

(21)申請案號：103101884

(22)申請日：中華民國 103 (2014) 年 01 月 17 日

(51)Int. Cl.：

G06F13/16 (2006.01)

G06F13/36 (2006.01)

G06F5/06 (2006.01)

(30)優先權：2013/02/04

美國

13/758,853

(71)申請人：L S I 公司 (美國) LSI CORPORATION

(US)

美國

(72)發明人：伯特 盧卡 BERT, LUCA (US)；席蒙內司克 霍利亞 SIMIONESCU, HORIA

(US)；巴德帝倪 雅南特 BADERDINNI, ANANT (IN)；伊許 馬克 ISH, MARK

(US)

(74)代理人：陳長文

申請實體審查：無 申請專利範圍項數：19 項 圖式數：5 共 30 頁

(54)名稱

用於藉由使用一命令推動模型減少一資料儲存系統中之寫入延遲之方法及系統

METHOD AND SYSTEM FOR REDUCING WRITE LATENCY IN A DATA STORAGE SYSTEM BY USING A COMMAND-PUSH MODEL

(57)摘要

本發明揭示一種資料儲存系統，其實施減少延遲之一命令推動模型。主機系統存取記憶體控制器之一非揮發性記憶體(NVM)裝置，以容許該主機系統將命令推動至定位於該 NVM 裝置中之一命令佇列中。該主機系統在不需要該記憶體控制器介入之情況下完成各 IO，從而消除該主機系統與該記憶體控制器之間之同步或交握之需要。對於寫入命令，記憶體控制器並不需要在完成該命令之後發佈一完成中斷至該主機系統，此係因為該主機系統在該寫入命令被推動至該記憶體控制器之該佇列中時認為該寫入命令已完成。全部此等特徵之組合導致總延遲之大量減少。

發明摘要

※ 申請案號：103/01892

※ 申請日：103.1.17

※IPC 分類：G06F 13/16 (2006.01),
G06F 13/36 (2006.01),
G06F 4/06 (2006.01).

【發明名稱】

用於藉由使用一命令推動模型減少一資料儲存系統中之寫入延遲之方法及系統

METHOD AND SYSTEM FOR REDUCING WRITE LATENCY IN
A DATA STORAGE SYSTEM BY USING A COMMAND-PUSH
MODEL

【中文】

本發明揭示一種資料儲存系統，其實施減少延遲之一命令推動模型。主機系統存取記憶體控制器之一非揮發性記憶體(NVM)裝置，以容許該主機系統將命令推動至定位於該NVM裝置中之一命令佇列中。該主機系統在不需該記憶體控制器介入之情況下完成各IO，從而消除該主機系統與該記憶體控制器之間之同步或交握之需要。對於寫入命令，記憶體控制器並不需要在完成該命令之後發佈一完成中斷至該主機系統，此係因為該主機系統在該寫入命令被推動至該記憶體控制器之該佇列中時認為該寫入命令已完成。全部此等特徵之組合導致總延遲之大量減少。

【英文】

A data storage system is provided that implements a command-push model that reduces latencies. The host system has access to a nonvolatile memory (NVM) device of the memory controller to allow the host system to push commands into a command queue located in the NVM device. The host system completes each IO without the need for intervention from the memory controller, thereby obviating the need for synchronization, or handshaking, between the host system and the memory controller. For write commands, the memory controller does not need to issue a completion interrupt to the host system upon completion of the command because the host system considers the write command completed at the time that the write command is pushed into the queue of the memory controller. The combination of all of these features results in a large reduction in overall latency.

【代表圖】

【本案指定代表圖】：第（ 3 ）圖。

【本代表圖之符號簡單說明】：

200	資料儲存系統/系統
220	主機系統
230	系統中央處理單元(CPU)/中央處理單元(CPU)
240	直接記憶體存取(DMA)引擎
250	記憶體驅動器
260	系統記憶體裝置
265	周邊互連(PCI)/周邊互連高速(PCIe)匯流排
270	記憶體控制器
280	控制器中央處理單元(CPU)/中央處理單元(CPU)
290	非揮發性記憶體(NVM)裝置
291	命令佇列/佇列
310	輸入/輸出(I/O)介面裝置
320	固態驅動機(SSD)
320a	實體磁碟(PD)

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：

無

發明專利說明書

(本說明書格式、順序，請勿任意更動)

【發明名稱】

用於藉由使用一命令推動模型減少一資料儲存系統中之寫入延遲之方法及系統

METHOD AND SYSTEM FOR REDUCING WRITE LATENCY IN A DATA STORAGE SYSTEM BY USING A COMMAND-PUSH MODEL

【技術領域】

本發明大體上係關於資料儲存系統，且更特定言之，本發明係關於一種用於在一資料儲存系統中使用一命令推動模型以減少寫入延遲之方法及系統。

【先前技術】

一儲存陣列或磁碟陣列係一資料儲存裝置，其包含多個硬碟機(HDD)或類似之永久儲存單元。一儲存陣列可容許依一有效率之方式儲存大量資料。可將一伺服器或工作站直接附接至該儲存陣列，使得該儲存陣列對於該伺服器或工作站而言係本端的。在該伺服器或工作站被直接附接至該儲存陣列之情況下，該儲存陣列通常稱為一直接附接儲存(DAS)系統。或者，可經由一儲存陣列網路(SAN)將一伺服器或工作站遠端地附接至該儲存陣列。在SAN系統中，儘管儲存陣列對於該伺服器或工作站而言不是本端的，但該陣列之磁碟驅動機對於該伺服器或工作站之作業系統(OS)而言似乎係本端附接的。

圖1繪示實施一命令拉動模型之一典型資料儲存系統2之一方塊圖。系統2包含一主機系統3、一記憶體控制器4及一周邊互連(PCI)或PCI高速(PCIe)匯流排5。控制器4包含一中央處理單元(CPU) 6、一記

憶體裝置7及一I/O介面裝置8。I/O介面裝置8經組態以依照已知資料傳送協定標準(諸如串行連接SCSI(SAS)標準、串行高級技術附件(SATA)標準或非揮發性記憶體主控制器介面高速(NVMe)標準)而執行資料傳送。I/O介面裝置8控制資料至多個實體磁碟(PD) 9之傳送或資料自多個實體磁碟(PD) 9之傳送。記憶體控制器4經由PCI匯流排5與一系統CPU 11及一系統記憶體裝置12通信。系統記憶體裝置12儲存由系統CPU 11執行之軟體程式及資料。系統記憶體裝置12之一部分用作為一命令佇列13。

在一典型寫入動作期間，系統CPU 11運行一記憶體驅動器軟體堆疊14，其將命令及資料儲存於命令佇列13中。當記憶體驅動器14將一命令儲存於命令佇列13中時，其通知記憶體控制器4：一命令已準備好被執行。當控制器CPU 6準備執行一命令時，其經由匯流排5自系統佇列13拉動命令或多個命令及相關聯之資料且向主機系統3發佈一完成中斷。當記憶體控制器4執行命令時，控制器CPU 6致使與該等命令相關聯之資料被暫時地儲存於控制器記憶體裝置7中且接著隨後經由I/O介面裝置8被寫入至PD 9之一或多者。

歷史上，已就每秒輸入/輸出(IO)(IOPS)且在一些情況中就每秒百萬位元組 (MB/s)而言量測圖1中所展示之類型之一基於HDD系統之效能。此一儲存系統之延遲通常給定為： $\text{Latency_Overall} = \text{Latency_SW_Stack} + \text{Latency_Controller} + \text{Latency_HDD} = 1/\text{IOPS}$ ，其中Latency_Overall為系統之總延遲，Latency_SW_Stack為與記憶體驅動器14相關聯之延遲，Latency_Controller為與記憶體控制器4相關聯之延遲，且Latency_HDD為與PD 9相關聯之延遲。Latency_SW_Stack之數量級通常為微秒(10^{-6} 秒)。同樣地，Latency_Controller之數量級通常為數十微秒。然而，Latency_HDD之數量級通常為毫秒(10^{-3} 秒)或數十毫秒。總延遲之大約99%係歸因於該等HDD之極緩慢之機械部件。因此，出於實

際目的，當判定系統效能時可忽略Latency_SW_Stack及Latency_Controller。換言之，系統效能可估計為等於Latency_HDD。

近來，存在自使用磁性HDD作為PD 9至使用固態驅動機(SSD)或SSD及HDD之一組合作為PD 9之一轉變。在業界中，使用基於SSD之解決方案被視為基於HDD之解決方案之一演進。然而，基於SSD之解決方案比基於HDD之解決方案快大約100倍且消耗更少功率。此基於SSD之解決方案此觀點已引起業界繼續使用基於SSD之解決方案中之預先存在的上述拉動方法學，來自命令佇列拉動命令至記憶體控制器中。並且，由於基於SSD之解決方案已在業界中被僅僅視為基於HDD之解決方案之一演進，因此已使用IOPS作為用於在實施基於SSD之解決方案之儲存系統中量測系統效能之效能量度。

然而，基於SSD之解決方案與基於HDD之解決方案之間之差異比其所呈現的更多，且傳統量度不應被用以量測實施基於SSD之解決方案之系統之效能。在實施一基於SSD之解決方案之一系統中，儲存系統之總延遲給定為： $\text{Latency_Overall} = \text{Latency_SW_Stack} + \text{Latency_Controller} + \text{Latency_SSD} = 1/\text{IOPS}$ ，其中Latency_SW_Stack為與記憶體驅動器14相關聯之延遲，Latency_Controller為與記憶體控制器4相關聯之延遲且Latency_SSD為與用作PD 9之SSD相關聯之延遲。不同於HDD之延遲，SSD之延遲之數量級為數十至數百微秒，例如，大體上在100至300微秒之範圍中，且與記憶體驅動器14及記憶體控制器4相關聯之延遲添加遠多於數十至數百微秒之延遲至總延遲。因此，在計算實施一基於SSD之解決方案之儲存系統之總延遲時，不應再忽視Latency_SW及Latency_Controller。

命令拉動方法需要記憶體驅動器14與記憶體控制器4之間之相當多之互動。此在基於HDD之系統中係一便利方法，此係因為其容許主機系統3之快速作業系統(OS)側幾乎完全地獨立於較緩慢之基於HDD

之控制器側，使得該OS側可在佇列13中堆積儘可能多之命令以提供更大佇列深度(QD)，此在基於HDD之解決方案中極為合意且普遍。接著，記憶體控制器4可以其自身之速度自佇列13拉動命令。儘管此方法在基於HDD之解決方案中極為便利，但歸因於記憶體控制器4與主機系統3之間所需之同步且歸因於記憶體控制器4可能在遠遠遲於命令被發佈之時間處撿取命令之事實，其添加大量額外延遲。此外，若存在大量工作待由記憶體控制器4來做(此為常有之情況)，則全部命令處理必須與記憶體控制器4之剩餘工作負荷競爭，此添加更多延遲至總延遲。

上述命令拉動模型對於基於HDD之解決方案極其有效，其中鑑於該方法提供之其他優點，添加50至500微秒至通常可能耗費約10,000微秒來完成之一命令係可忽略的。然而，當用於實施一基於SSD之解決方案之一儲存系統中時(其中存取時間可低至100微秒)或在一動態隨機存取記憶體(DRAM)寫回(WB)緩衝器用於記憶體控制器4中之情況下(低至1至5微秒)，該命令拉動模型並未產生可接受之結果。

然而，如上文所指示，實施基於SSD之解決方案之儲存系統之總延遲仍被估計為等於Latency_SSD，同時使得Latency_SW_Stack及Latency_Controller在估計之外。由於此原因，減少實施基於SSD之解決方案之儲存系統之總延遲之嘗試已首先集中於減少Latency_SSD而非減少Latency_SW_Stack或Latency_Controller。

存在對實施一基於SSD之解決方案且藉由顯著減少Latency_SW_Stack及Latency_Controller之一或兩者來顯著減少總延遲之一儲存系統一需要。

【發明內容】

本發明提供一資料儲存系統、一記憶體控制器、一方法及一電

腦可讀媒體，其等全部實施減少延遲之一命令推動模型。該資料儲存系統包括一主機系統、一記憶體控制器、至少一SSD及互連該主機系統及該記憶體控制器之一匯流排。該主機系統包括一系統處理器及一系統記憶體裝置。該記憶體控制器包括一控制器處理器、一非揮發性記憶體(NVM)裝置及一輸入/輸出(I/O)介面裝置。該NVM裝置之一部分用作一命令佇列。該SSD連接至該I/O介面裝置且經組態為一PD陣列。該主機系統經由該匯流排存取該NVM裝置且經由該匯流排將命令推動至該NVM裝置之該命令佇列中。

該方法包括：

在包括一控制器處理器、一NVM裝置及連接至經組態為一PD陣列之至少一SSD之一I/O介面裝置之一記憶體控制器中，將該NVM裝置之一部分組態為一命令佇列；及

在一主機系統經由一匯流排與該記憶體控制器互連之情況下，經由該匯流排將一命令推動至該記憶體控制器中；及

在該記憶體控制器中，將該命令儲存於該NVM裝置之該命令佇列中。

該電腦可讀媒體具有儲存於其上之一電腦程式，該電腦程式由一主機系統之一處理器執行以將命令推動至經由一匯流排連接至該主機系統之一記憶體控制器之一NVM裝置之一命令佇列中。該電腦程式包括第一程式碼部分及第二程式碼部分。該第一程式碼部分接收讀取或寫入經組態為一PD陣列之一SSD之一或多個位址之一命令。該第二程式碼部分將該命令推動至該記憶體控制器中。根據一闡釋性實施例，該第二程式碼部分將該命令推動至該記憶體控制器之該NVM裝置之該命令佇列中。根據另一闡釋性實施例，該第二程式碼部分將該命令推動至該記憶體控制器之一直接記憶體存取(DMA)引擎中，該直接記憶體存取(DMA)引擎將該命令儲存於該記憶體控制器之該NVM

裝置之該命令佇列中。

將自下列描述、圖式及隨附申請專利範圍明白本發明之此等及其他特徵及優點。

【圖式簡單說明】

圖1繪示實施一命令拉動模型之一已知資料儲存系統之一方塊圖。

圖2繪示根據一闡釋性實施例之實施一命令推動模型之一資料儲存系統之一方塊圖。

圖3繪示根據另一闡釋性實施例之實施一命令推動模型之一資料儲存系統之一方塊圖。

圖4繪示表示由圖2中所展示之資料儲存系統執行之命令推動方法之一流程圖。

圖5繪示表示由圖3中所展示之資料儲存系統執行之命令推動方法之一流程圖。

【實施方式】

根據本發明，提供實施一命令推動模型之一資料儲存系統，該命令推動模型減少與記憶體驅動器及記憶體控制器相關聯之延遲。該命令推動模型消除了對用於寫入命令IO之主機系統與記憶體控制器之間之同步之需要，從而消除了與同步相關聯之延遲。當在該主機系統中發佈命令IO，該主機系統存取該記憶體控制器之一記憶體裝置且將該等命令IO及相關聯之資料推動至該記憶體裝置之一命令佇列中。因此，該主機系統在無該記憶體控制器之任何介入之情況下完成各寫入命令IO，從而消除該主機系統與該記憶體控制器之間之同步或交握之需要。

該記憶體控制器之記憶體裝置經保護而免於電源故障及典型記憶體錯誤，且就此而言可視為一永久儲存元件。因此，該主機系統可

自其將寫入命令及任何相關聯資料推動至該記憶體控制器之該記憶體裝置之該命令佇列中時起認為一寫入命令IO已安全完成。換言之，自該主機系統之角度看，當其將該寫入命令推動至該記憶體控制器之該記憶體裝置之該命令佇列中時，就如同其已完成至PD 9之一寫入。因此，該主機系統在無該記憶體控制器之任何介入之情況下完成各寫入命令IO，從而消除該主機系統與該記憶體控制器之間之同步或交握之需要。

此外，與上文參考圖1所描述之命令推動模型相比，本發明之記憶體控制器在其已完成對該寫入命令之處理之後並不需要發佈一完成中斷至該主機系統，此係因為該主機系統在其將該寫入命令推動至該記憶體控制器之該命令佇列中時認為寫入命令已完成。此特徵容許該主機系統及該記憶體控制器可相對於寫入命令彼此獨立地操作。全部此等特徵之組合導致Latency_SW_Stack及Latency_Controller兩者之大量減少，此導致總延遲之大量減少。

因此，該主機系統將命令及任何相關聯資料推動至定位於該記憶體控制器之該記憶體裝置中之該命令佇列中，而非使用上文參考圖1描述的其中命令在系統記憶體12之命令佇列13中準備且隨後由記憶體控制器4拉入且處理之命令拉動模型。可依若干方式進行該主機系統將該等命令及任何相關聯資料推動至該記憶體控制器之該記憶體裝置之該命令佇列中之方法。根據一第一闡釋性實施例，該主機系統之記憶體驅動器程式化地將命令及相關聯資料推動至該記憶體控制器之該記憶體裝置之該命令佇列中。根據本文中所描述之一第二闡釋性實施例，該記憶體控制器之一直接記憶體存取(DMA)引擎將命令及任何相關聯資料推動至該記憶體控制器之該記憶體裝置之該命令佇列中。此等實施例之兩者消除該主機系統與該記憶體控制器之間之同步或交握之需要。

圖2繪示根據該第一闡釋性實施例之實施一命令推動模型之一資料儲存系統20之一方塊圖。系統20包含一主機系統30、一記憶體控制器70、一周邊互連(PCI)或PCI高速(PCIe)匯流排65及包括PD 120a之至少一SSD 120。主機系統30包含一系統CPU 40及一系統記憶體裝置60。記憶體控制器70包含一CPU 80、一非揮發性記憶體(NVM)裝置90及一I/O介面裝置110。NVM裝置90通常為一DRAM裝置。NVM裝置90之一部分用作一命令佇列91。I/O介面裝置110經組態以依照已知資料傳送協定(諸如SAS、SATA及/或NVMe標準)執行資料傳送。I/O介面裝置110控制資料至SSD 120之多個PD之傳送及資料自SSD 120之多個PD之傳送。記憶體控制器70經由PCI匯流排65及記憶體驅動器50與系統CPU 40通信。系統記憶體裝置60儲存由系統CPU 40執行之軟體程式及資料。

根據此闡釋性實施例，記憶體驅動器50經由匯流排65直接與NVM裝置90通信。在一典型寫入動作期間，系統CPU 40運行包含記憶體驅動器50之一軟體堆疊，其將命令及資料推動至NVM裝置90之命令佇列91中且通知控制器CPU 80：該命令及資料位於佇列91中。主機系統30與記憶體控制器70之間不需要同步或交握來執行此推動動作。一旦記憶體驅動器50已將一寫入命令及相關聯資料推動至命令佇列91中，主機系統30之OS即認為該寫入命令已完成。因此，記憶體控制器70不必發佈一完成中斷至主機系統30以告知主機CPU 40：已完成該寫入命令。

當控制器CPU 80準備執行該命令時，其自NVM裝置90擷取該命令及資料且執行該命令。在一寫入命令之情況下，控制器CPU 80引起相關聯之資料經由I/O介面裝置110被寫入至SSD PD 120a。在一讀取命令之情況下，控制器CPU 80檢查NVM 90之快取區(為清晰起見圖中未展示)以判定待讀取之資料是否位於快取區中，且若該資料位於

快取區中，則自快取區讀取該資料且將其傳送至主機系統30。若待讀取之資料未位於快取區中，則CPU 80致使該資料被自SSD PD 120a讀取且傳送至主機系統30。

圖2之命令推動模型與圖1之命令拉動模型之間之關鍵差異之一者在於：命令所有權甚至在控制器CPU 80感知該命令之前即被自主機系統30轉移至NVM裝置90之佇列91。在圖1之命令拉動模型中，與一寫入命令相關聯之命令所有權直至主機系統3自記憶體控制器4接收完成中斷才被轉移。此外，在圖1之命令拉動模型中，控制器CPU 6與主機系統3之間發生同步或交握以將命令自系統記憶體裝置12之佇列13拉動至記憶體控制器4中。相較於圖1中所展示之命令拉動模型，消除此等延遲通常將導致寫入延遲減少一或兩個數量級。

由於在命令被推動至NVM 90之佇列91中之後記憶體控制器70即取得命令之所有權，所以保護NVM 90免於電源故障係重要的。可依若干方式達成此保護，而一種方式係將一備用電池(BBU)、超電容器(Supercap)動態隨機存取記憶體(DRAM)裝置用於NVM 90。熟習此項技術者將瞭解可出於此目選擇一BBU、Supercap DRAM裝置或其他合適記憶體裝置之方式。

為了使記憶體驅動器50有效率地將命令推動至佇列91中，NVM裝置90應由記憶體驅動器50經由匯流排65直接存取。可依若干方式完成此可存取性。完成此可存取性之一種方法係在NVM裝置90中使用主機系統30中所使用之相同命令及資料結構。如此項技術者將瞭解，通過對基位址暫存器(BAR)之一便利程式化，此在PCIe中係可行的。可藉由記憶體驅動器50程式化地完成自主機系統3至NVM裝置90之資料傳送。

記憶體驅動器50亦應具備判定NVM裝置90中哪一個條目可用以使用命令資訊對其等進行映射/標記之一方式。各命令由一資料集及

命令資訊組成。該命令資訊包含諸如讀取/寫入、塊位址、指標及旗標之資訊。該命令資訊根據命令類型而變動且獨立於該資料集而產生。需要將該資料集及該命令資訊兩者移動至NVM 90中且必須以對應命令資訊對該資料集進行映射/標記以確保該命令資訊與該資料集之間之一致性。完成此之方式將取決於所使用之記憶體控制器70之類型或組態而變動。所有記憶體控制器可經組態以提供此功能性，但該解決方案之效率將取決於記憶體控制器之硬體、軟體及/或韌體實施方案。因此，此問題將需要根據個案具體情況來處理，如熟習此項技術者將瞭解。

圖3繪示根據第二闡釋性實施例之一資料儲存系統200之一方塊圖，其中使用一DMA引擎將命令及資料推動至記憶體控制器之記憶體裝置之佇列中。儘管DMA引擎可為主機系統之一部分或記憶體控制器之一部分，但出於闡釋性目的，其經展示及描述為記憶體控制器之一部分。系統200包含一主機系統220、一記憶體控制器270、一PCI或PCIe匯流排265及包括PD 320a之至少一SSD 320。主機系統220包含一系統CPU 230及一系統記憶體裝置260。記憶體控制器270包含一DMA引擎240、一CPU 280、一NVM裝置290及一I/O介面裝置310。通常，NVM裝置290為一DRAM裝置，且其具有與上文參考NVM 90所描述相同之保護。NVM裝置290之一部分用作一命令佇列291。I/O介面裝置310經組態以依照已知資料傳送協定標準(諸如SAS、SATA及/或NVMe標準)執行資料傳送。I/O介面裝置310控制資料至SSD 320之多個PD 320a之傳送或資料自SSD 320之多個PD 320a之傳送。記憶體控制器270經由PCI匯流排265與系統CPU 230通信且與記憶體驅動器250通信。系統記憶體裝置260儲存由系統CPU 230執行之軟體程式及資料。

根據此闡釋性實施例，DMA引擎240經由匯流排265與記憶體驅

動器250通信且直接與NVM裝置290通信。在一典型寫入動作期間，記憶體驅動器250經由匯流排265將命令及任何相關聯之資料推動至DMA引擎240中。DMA引擎240將命令及資料儲存於NVM裝置290之命令佇列291中。主機系統220與記憶體控制器270之間不需要同步或交握來執行推動動作。一旦DMA引擎240已將一寫入命令及相關聯資料儲存至命令佇列291中，較佳地其即告知(通常藉由發佈一中斷)控制器CPU 280：該命令及資料位於NVM 290中。主機系統220之OS自該命令及資料被傳送至DMA引擎240時起即認為該寫入命令已完成。因此，記憶體控制器270無需發佈一完成中斷至主機系統220來告知其：已完成該寫入命令。

當控制器CPU 280準備執行該命令時，其自NVM裝置290擷取該命令且執行該命令。在一寫入命令之情況下，控制器CPU 280致使相關聯之資料經由I/O介面裝置310被寫入至SSD PD 320a中。在一讀取命令之情況下，控制器CPU 280檢查NVM 290之快取區(為清楚起見圖中未展示)以判定待讀取之資料是否位於快取區中，且若其位於快取區中，則自快取區讀取資料且將其傳送至主機系統220。若待讀取之資料未位於快取區中，則CPU 280致使資料被自SSD PD 320a讀取且傳送至主機系統220。

如同上文參考圖2所描述之命令推動模型，以圖3之命令推動模型，命令所有權甚至在控制器CPU 280感知命令之前就被自主機系統220傳送至NVM裝置290之佇列291。如同上文參考圖2所描述之命令推動模型，以圖3之命令推動模型，控制器CPU 280與主機系統22之間不需要同步或交握以將命令推動至佇列291中。因此，消除了上文參考圖1所描述之延遲。

圖4繪示表示部分由圖2中所展示之主機系統30且部分由圖2中所展示之記憶體控制器70執行之根據一闡釋性實施例之命令推動方法之

一流程圖。儘管亦可針對多個命令同時執行該方法之步驟，但將參考一單個命令描述該方法。如方塊401所指示，系統CPU 40發佈一讀取或寫入命令。主機系統30之記憶體驅動器50將該命令推動至NVM裝置90之佇列91中且通知記憶體控制器70：該命令位於NVM裝置90中，如方塊402所指示。

在一寫入命令之情況下，亦將與該命令相關聯之資料推動至NVM裝置90中。隨後，將自NVM裝置90讀取之命令及資料寫入至SSD 120之PD 120a。在已被推動至NVM裝置90中之一讀取命令之情況下，記憶體控制器70通常將判定資料是否儲存於該NVM裝置90之快取區中。若該記憶體控制器70判定該資料儲存於快取區中(即，一快取命中)，則其自快取區讀取資料且將其傳送至主機系統30。若記憶體控制器70判定該資料未儲存於快取區中(即，一快取未中)，則其自SSD 120之PD 120a讀取資料且將該資料傳送至主機系統30。

在其中一寫入命令被推動至NVM裝置90之佇列91中之情況下，一旦記憶體驅動器50將寫入命令推動至佇列91中，IO即在主機系統30中完成。在其中一讀取命令被推動至佇列91中之情況下，IO直至該讀取資料被返回至主機系統30才在主機系統30中完成，此係在圖1中所展示之已知儲存系統2中處理讀取命令之方式。歸因於不再需要同步以將寫入命令自主機系統30傳送至記憶體控制器70之事實及歸因於主機系統30不再必須在等待一完成中斷之後認為寫入命令完成之事實，相對於寫入命令，由本發明之方法及系統所達成之延遲減少最為顯著。

圖5繪示表示部分由圖3中所展示之主機系統220且部分由圖3中所展示之記憶體控制器270所執行之根據一闡釋性實施例之命令推動方法之一流程圖。儘管亦可針對多個命令同時執行該方法之步驟，但將參考一單個命令描述該方法。系統CPU 230發佈一讀取或寫入命

令，如方塊501所指示。記憶體控制器250將該命令推動至DMA引擎240，如方塊502所指示。DMA引擎240將命令儲存於NVM裝置290之佇列291中且通知記憶體控制器270：命令位於NVM裝置290中，如方塊503所指示。相對於通知記憶體控制器270一命令已被放置於佇列291中之方式，本發明係不受限的。可依若干方式完成此通知，例如，發佈一中斷、更新暫存器中之一值等等，如熟習此項技術者將瞭解。

主機系統30及220之功能性及記憶體控制器70及270之功能性可以硬體、軟體、韌體或其等之一組合實施。用於以軟體或硬體實施功能性之電腦程式碼儲存於一電腦可讀媒體(CRM)上，諸如系統記憶體裝置60及260或NVM裝置90或290或一些其他記憶體裝置。該CRM可為任何類型之記憶體裝置，包括但不限於磁性儲存裝置、固態儲存裝置、快閃記憶體裝置及光學儲存裝置。通常，CPU 40、80、230及280之各者包括至少一微處理器，但可包括能夠提供執行相關聯任務所必需或所需要之功能性之任何類型之處理器，包含(例如)一微處理器、一數位信號處理器(DSP)、一專用積體電路(ASIC)及一芯片上之一系統(SOC)。如本文中所使用，術語「處理器」意欲表示可經程式化或組態以執行上文所描述之任務及被視為容許CPU 40、80、230及280執行其等作用所必需之任何額外任務。

應注意：出於演示本發明之原理及概念之目的，已參考若干闡釋性或例示性實施例描述本發明。如熟習此項技術者將明白，在不脫離本發明之範疇之情況下可對上述闡釋性實施例作出許多變動。全部此等變動係在本發明之範疇內。例如，儘管PD 120a及320a分別在圖2及圖3中展示為分別僅以至少一SSD 120或320實施，但PD 120a及320a可以一或多個SSD及一或多個HDD之一組合實施。

【符號說明】

2	資料儲存系統/系統
3	主機系統
4	記憶體控制器
5	周邊互連(PCI)/PCI高速(PCIe)匯流排
6	中央處理單元(CPU)
7	記憶體裝置
8	輸入/輸出(I/O)介面裝置
9	實體磁碟(PD)
11	系統中央處理單元(CPU)
12	系統記憶體裝置
13	命令佇列/系統佇列/佇列
14	記憶體驅動器軟體堆疊/記憶體驅動器
20	資料儲存系統
30	主機系統
40	系統中央處理單元(CPU)/主機中央處理單元(CPU)/ 中央處理單元(CPU)
50	記憶體驅動器
60	系統記憶體裝置
65	周邊互連(PCI)/周邊互連高速(PCIe)匯流排
70	記憶體控制器
80	控制器中央處理單元(CPU)/ 中央處理單元(CPU)
90	非揮發性記憶體(NVM)裝置
91	命令佇列/佇列
110	輸入/輸出(I/O)介面裝置
120	固態驅動機(SSD)
120a	實體磁碟(PD)

200	資料儲存系統/系統
220	主機系統
230	系統中央處理單元(CPU)/中央處理單元(CPU)
240	直接記憶體存取(DMA)引擎
250	記憶體驅動器
260	系統記憶體裝置
265	周邊互連(PCI)/周邊互連高速(PCIe)匯流排
270	記憶體控制器
280	控制器中央處理單元(CPU)/中央處理單元(CPU)
290	非揮發性記憶體(NVM)裝置
291	命令佇列/佇列
310	輸入/輸出(I/O)介面裝置
320	固態驅動機(SSD)
320a	實體磁碟(PD)

申請專利範圍

1. 一種資料儲存系統，其包括：

一主機系統，其包括一系統處理器及一系統記憶體裝置；

一記憶體控制器，其包括一控制器處理器、一非揮發性記憶體(NVM)裝置及一輸入/輸出(I/O)介面裝置，其中該NVM裝置之一部分用作一命令佇列；

至少一固態驅動機(SSD)，其連接至該I/O介面裝置，其中該至少一SSD經組態為一實體磁碟驅動機(PD)陣列；及

一匯流排，其將該主機系統與該記憶體控制器互連，其中該主機系統經由該匯流排存取該NVM裝置且經由該匯流排將命令推動至該NVM裝置之該命令佇列中。

2. 如請求項1之資料儲存系統，其中該系統處理器執行一記憶體驅動器程式，該記憶體驅動器程式經由該匯流排存取該NVM裝置且經由該匯流排將該等命令推動至該NVM裝置之該命令佇列中。

3. 如請求項1之資料儲存系統，其中該記憶體控制器進一步包括直接存取該NVM裝置之一直接記憶體存取(DMA)引擎，且其中該系統處理器執行一記憶體驅動器程式，該記憶體驅動器程式經由該匯流排將該等命令推動至該DMA引擎中，且其中該DMA引擎將該等命令儲存於該NVM裝置之該命令佇列中。

4. 如請求項1之資料儲存系統，其中該主機系統在已將一命令推動至該NVM裝置之該命令佇列中時通知該記憶體控制器。

5. 如請求項2之資料儲存系統，其中該主機系統及該NVM裝置使用一相同命令及資料結構。

6. 如請求項5之資料儲存系統，其中該匯流排係一周邊互連高速

(PCIe)匯流排，且其中該記憶體驅動器程式藉由使用該PCIe匯流排之基位址暫存器(BAR)而存取該NVM裝置。

7. 一種記憶體控制器，其包括：

一輸入/輸出(I/O)介面裝置，其中該NVM裝置之一部分用作一命令佇列；

至少一固態驅動機(SSD)，其連接至該I/O介面裝置，其中該至少一SSD經組態為一實體磁碟驅動機(PD)陣列；

一控制器處理器；及

一非揮發性記憶體(NVM)裝置，該NVM裝置之一部分經分配為一命令佇列，其中該NVM裝置經組態以由一主機系統經由互連該主機系統與該記憶體控制器之一匯流排存取，其中該NVM裝置經組態以容許該主機系統經由該匯流排將命令推動至該NVM裝置之該命令佇列中。

8. 如請求項7之記憶體控制器，其中該NVM裝置由該主機系統經由由該主機系統之一處理器執行之一記憶體驅動器程式而存取，且其中該匯流排係一周邊互連高速(PCIe)匯流排，且其中該NVM裝置可由該記憶體驅動器程式通過該PCIe匯流排之基位址暫存器(BAR)存取。

9. 如請求項7之記憶體控制器，其中該記憶體控制器進一步包括：

一直接記憶體存取(DMA)引擎，其經由該匯流排與該主機系統通信及直接與該NVM裝置通信，且其中該DMA引擎由該主機系統經由該匯流排存取，且該DMA引擎接收由該記憶體驅動器程式推動至該DMA引擎之命令且將該等命令儲存於該NVM裝置之該命令佇列中。

10. 如請求項8之記憶體控制器，其中該NVM裝置使用與該主機系統相同之一命令及資料結構。

11. 一種減少一資料儲存系統中之延遲之方法，該方法包括：

在包括一控制器處理器、非揮發性記憶體(NVM)裝置及連接至經組態為一實體磁碟驅動機(PD)陣列之至少一固態驅動機(SSD)之一輸入/輸出(I/O)介面裝置之記憶體控制器中，將該NVM裝置之一部分組態為一命令佇列；

在一主機系統經由一匯流排與該記憶體控制器互連之情況下，經由該匯流排將一命令推動至該記憶體控制器中；及

在該記憶體控制器中，將該命令儲存於該NVM裝置之該命令佇列中。

12. 如請求項11之方法，其中該主機系統之一系統處理器執行一記憶體驅動器程式，其經由該匯流排存取該NVM裝置且經由該匯流排將該等命令推動至該NVM裝置之該命令佇列中。

13. 如請求項11之方法，其中該記憶體控制器進一步包括一直接記憶體存取(DMA)引擎，且其中該主機系統之一系統處理器執行經由該匯流排將命令自該主機系統推動至該DMA引擎之一記憶體驅動器程式，且其中該DMA引擎經由該匯流排將該等命令儲存至該NVM裝置之該命令佇列中。

14. 如請求項12之方法，其進一步包括：

其中該主機系統在已將一命令推動至該NVM裝置之該命令佇列中時通知該記憶體控制器。

15. 如請求項12之方法，其中該主機系統及該NVM裝置使用一相同命令及資料結構。

16. 如請求項15之方法，其中該匯流排係一周邊互連高速(PCIe)匯流排，且其中該記憶體驅動器程式藉由使用該PCIe匯流排之基位址暫存器(BAR)而存取該NVM裝置。

17. 一種非暫時性電腦可讀媒體，其具有儲存於其上之一電腦程

式，該電腦程式由一主機系統之一處理器執行以將命令推動至經由一匯流排連接至該主機系統之一記憶體控制器之一非揮發性記憶體(NVM)裝置之一命令佇列中，該電腦程式包括：

一第一程式碼部分，其接收一命令以讀取或寫入經組態為一實體磁碟驅動機(PD)陣列之一固態驅動機(SSD)之一或多個位址；及

一第二程式碼部分，其將該命令推動至該記憶體控制器中。

18. 如請求項17之非暫時性電腦可讀媒體，其中該第二程式碼部分將該命令推動至該NVM裝置之該命令佇列中，且其中該電腦程式進一步包括：

一第三程式碼部分，其在該命令已被推動至該記憶體控制器之該NVM裝置之該命令佇列中時通知該記憶體控制器之一處理器。

19. 如請求項17之非暫時性電腦可讀媒體，其中該第二程式碼部分將該命令推動至該記憶體控制器之一直接記憶體存取(DMA)引擎中，其中該DMA引擎被直接連接至該NVM裝置以容許該DMA引擎將該命令儲存於該命令佇列中，且其中該電腦程式進一步包括：

一第三程式碼部分，其在該命令已被推動至該記憶體控制器之該NVM裝置之該命令佇列中時通知該記憶體控制器之一處理器。

圖式

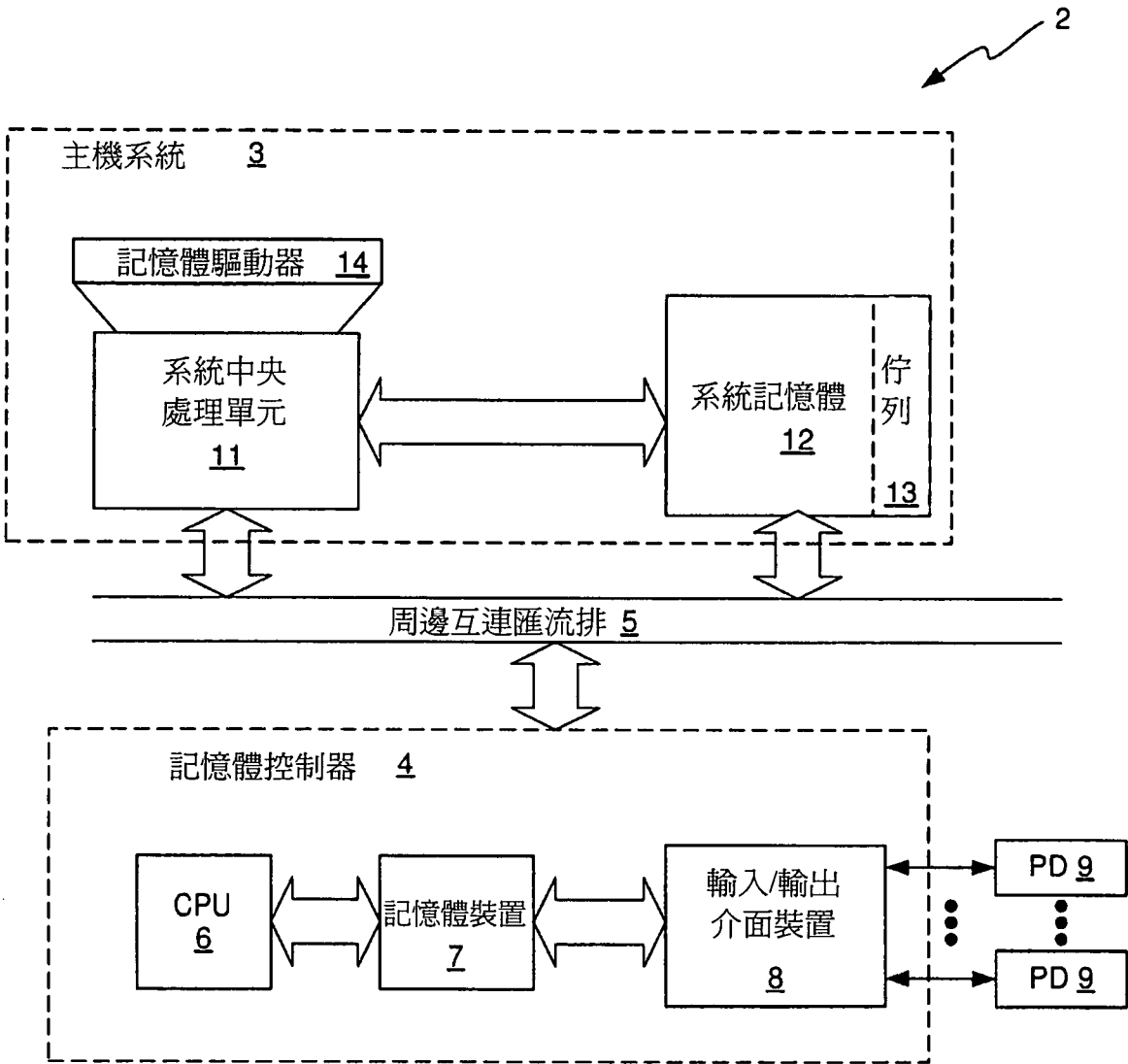


圖1

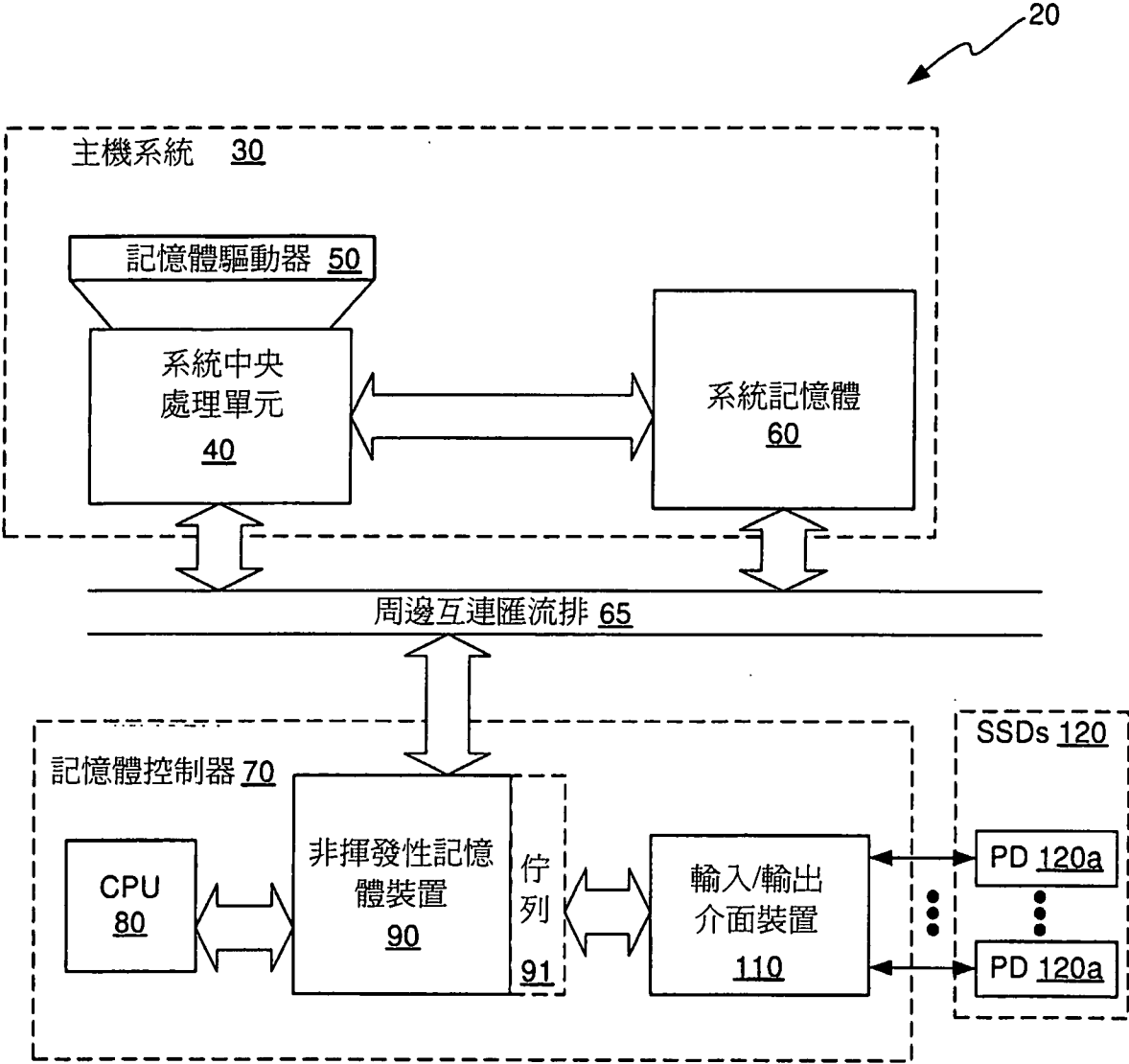


圖2

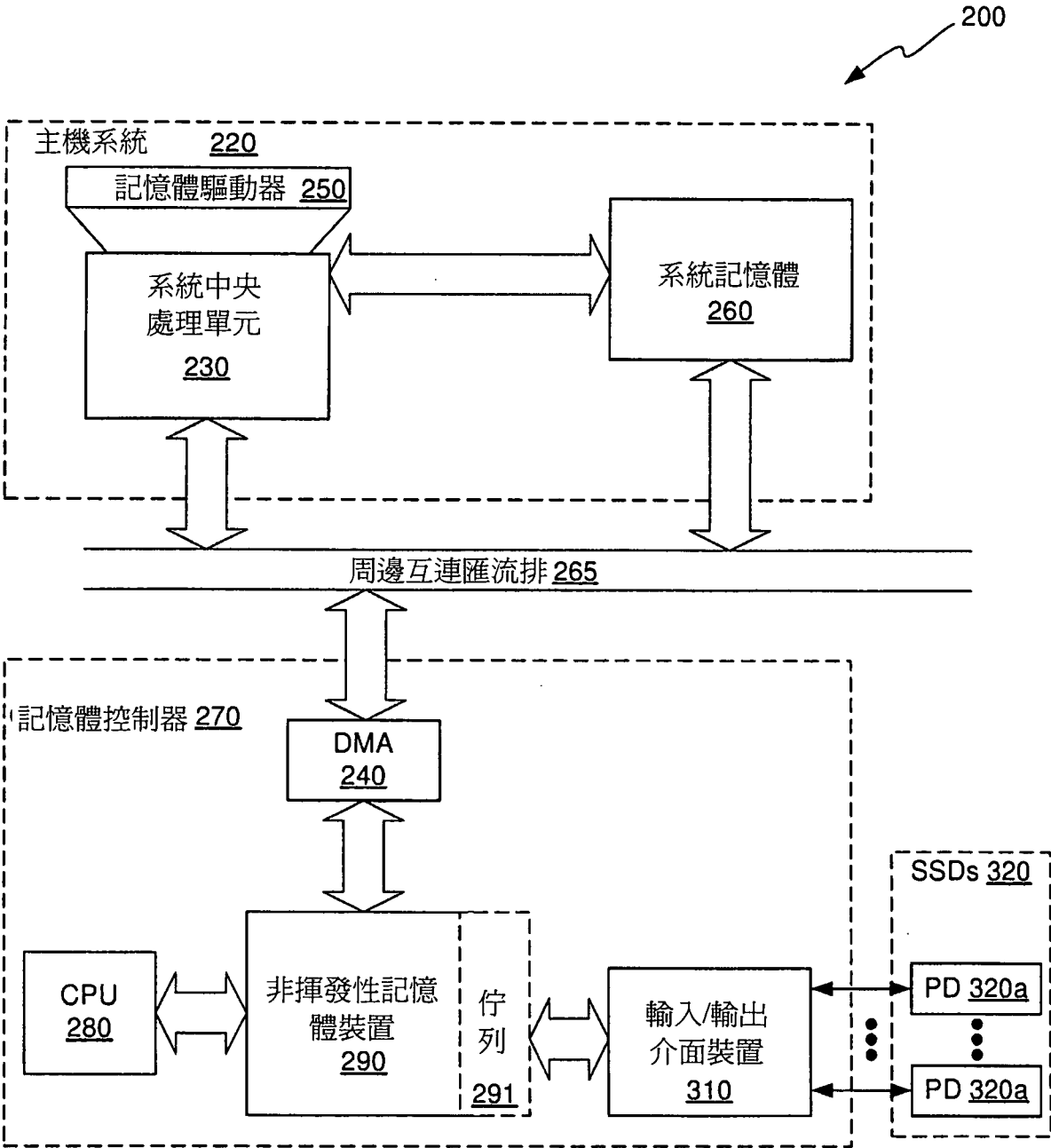


圖3

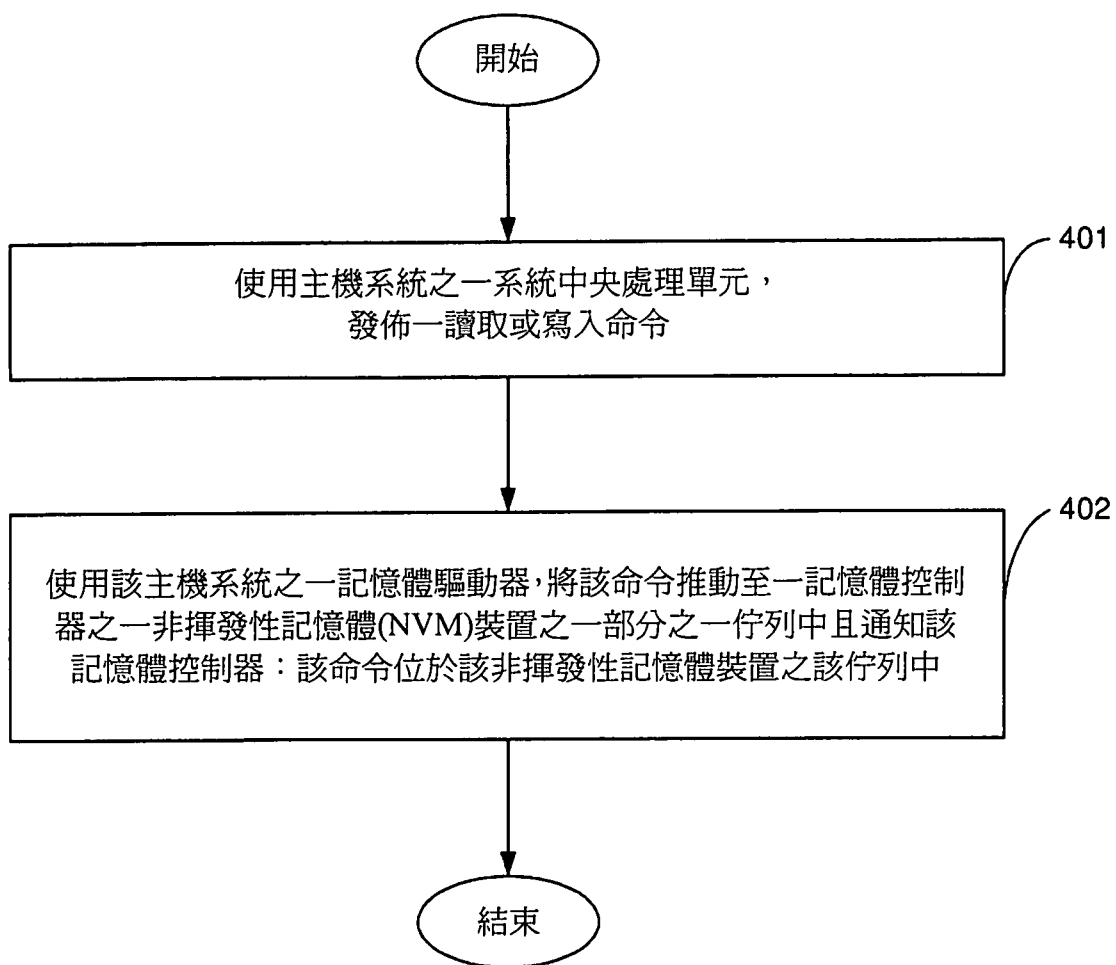


圖4

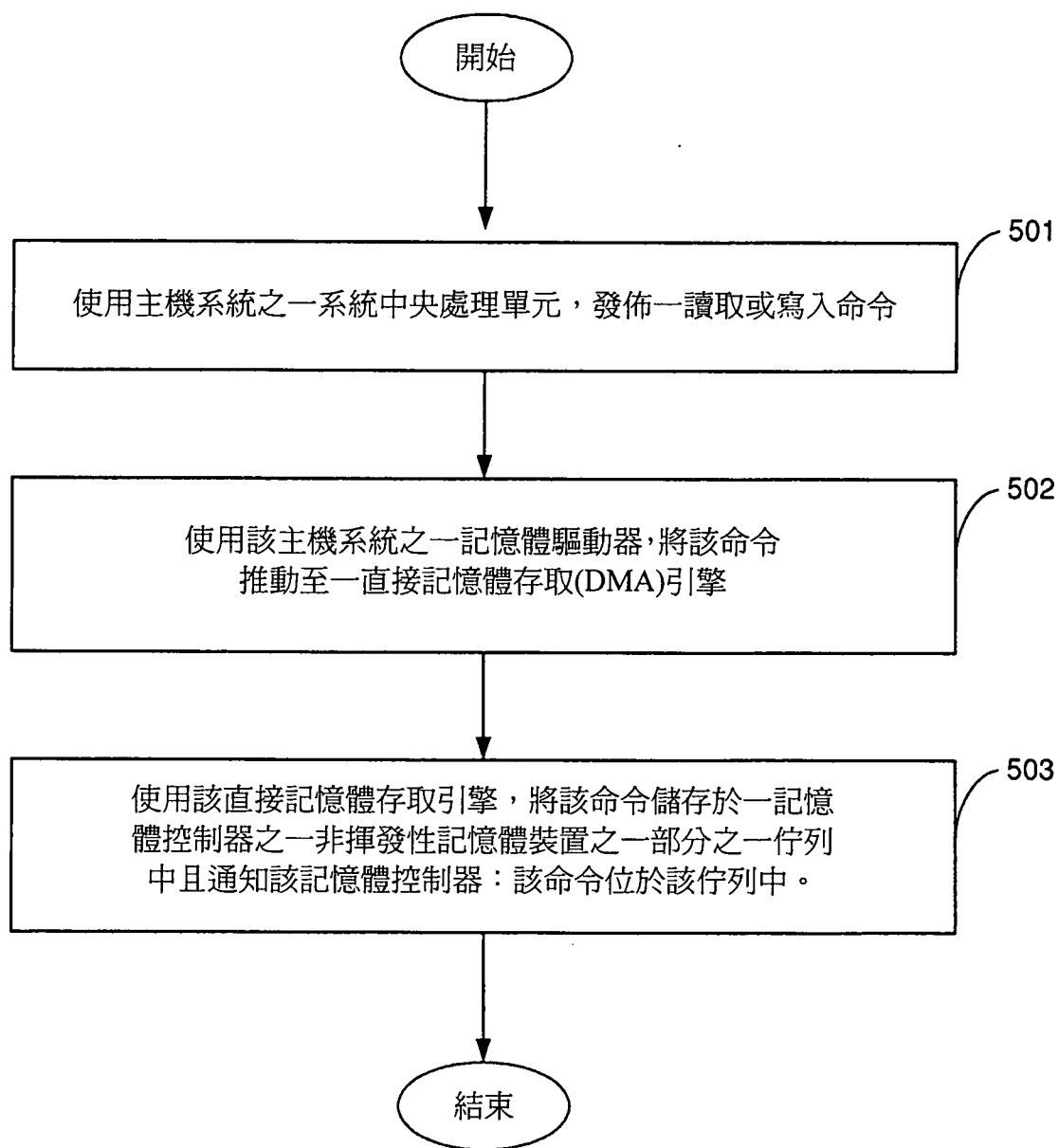


圖5