申請日期	00.7.11
案號	00116041
類別	G06F 11/36

（以上各欄由本局填註）

公告本
A4
C4

518463

發明 新型 專利說明書

一、發明新型名稱	中文	電腦免疫系統及在電腦系統中偵測不必要程式碼之方法
	英文	
二、發明創作人	姓名	彼得 A. J. 文德麥得
	國籍	荷蘭
	住、居所	澳洲 NSW 2106 紐波特海岸貝利爾卓依路 201 號
三、申請人	姓名（名稱）	vCIS 股份有限公司
	國籍	美國
	住、居所（事務所）	美國加州 90272 帕西翡克帕利塞德歐斯金路 522 號
	代表人姓名	羅伯特 密特羅

（由本局填寫）

承辦人代碼：	A6 B6
大　　類：	
ＩＰＣ分類：	

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	2000/07/14	60/218,489	
美國	2000/08/18	09/642,625	

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

五、發明說明（ | ）

本發明宣告美國暫時申請案號 60/218489 之優先權，其申請日爲西元 2000 年 7 月 17 日，其內容在此一併做爲參考。

本發明是有關於一種電腦防護領域，且特別是有關顯示包括比如電腦病毒與特洛伊病毒之惡意或自我散播行爲之電腦程式之偵測。

偵測病毒係個人電腦領域之重要的事。隨著通訊網路，如網際網路之成長與資料交換之增加，包括使用 E-mail 來通訊之快速成長，透過通訊或檔案交換之電腦病毒感染係明顯地增加。感染係以病毒形式出現，但一般係相關於電腦病毒，特洛伊程式或其他惡意程式碼之形式。由 E-mail 傳達病毒之事件不管在傳播速度與損壞程式上都相當令人注意，而 ISP 與公司也遭受著服務問題與 E-mail 能力之損失。在許多事件中，試著適當避免檔案交換或由 E-mail 傳達之病毒係大大地使電腦使用者不方便。因而，需要偵測與處理病毒攻擊之改善策略。

偵測病毒之一種傳統方法係掃描簽名。掃描簽名系統使用由已知惡意程式碼所獲得之範例程式碼圖樣，並掃描在其他程式程中出現之這些圖樣。在某些情況中，被掃描之程式碼係先透過仿效(emulate)而被分析，所得程式碼係掃描簽名或功能簽名。掃描簽名方法之主要限制是，只能偵測已知之惡意程式碼，也就是當感染時，只能辨認符合已知惡意程式碼之範例簽名。未預先辨認之所有病毒或惡意程式碼及在簽名資料庫之最後更新日後所產生之所有

五、發明說明（7）

病毒或惡意程式碼將不會被偵測。因此，新產生之病毒係不被此方法偵測；而且也無法偵測先前所獲得且包含於簽名資料庫內之被重寫簽名之程式碼之病毒。

此外，簽名分析技術無法辨認病毒之存在，如果簽名不在預期形式出現之程式碼內對準的話。另外，病毒之作者可藉由操作程式碼取代或藉由插入多餘或隨機程式碼至病毒功能中而模糊病毒之親認。係插入能改變病毒簽名之無意義程式碼，使得簽名掃描程式無法偵測，而不會減少病毒傳播與傳送其破壞能力。

另一種病毒偵測方法是完整(integrity)檢查。完整檢查系統由已知、良好應用程式碼中拉出程式碼範例。此程式碼範例係連同由如可執行程式表頭與檔案長度之程式檔案所得之資訊，以及範例之日期與時間而儲存。程式檔案係在資料庫定時檢查以確保程式檔案未被改變。如果使用者升級電腦之作業系統或安裝或升級應用軟體，完整檢查程式產生改變後檔案之長列表。根據完整檢查之病毒偵測系統之主要缺點是，當改變應用程式時，會出現許多的病程活動警告。對使用者而言，係很難決定，什麼時候警告係代表對電腦系統之標準攻擊。

計算值(checksum)監視系統係藉由產生各程式檔案之循環多餘檢查(CRC)值而偵測病毒。程式檔案之改變係由CRC 值改變而偵測。計算值監視對完整檢查系統之改良在於，惡意程式碼更難於使監視無效。另一方面，計算值監視具有與完整檢查系統相同之限制在於，會發出許多錯誤

五、發明說明（7）

警告，且難於辨認哪一種警告代表真正病毒或感染。

行爲攔截系統係藉由互動於目標電腦之作業系統與監視潛藏惡意行爲而偵測病程活動。當偵測此種惡意行爲時，此動作係被阻擋，且使用者係被通知將會發生潛藏危險活動。此潛藏惡意程式碼可由使用者允許來執行此行動。這使得行爲攔截系統在某些程度上是不可靠的，因爲系統之有效性係關於使用者之輸入。此外，常駐行爲攔截系統係有時被偵測且被惡意程式碼所失能。

另一種傳統偵測感染之方法係使用僞檔案。此方法一般用於合併其他病毒偵測方法以偵測現有並活動之感染。這代表，惡意程式碼係正在執行於標的電腦上，且正在改變檔案。當僞檔案被改變時，代表病毒被偵測。許多病毒係知道僞檔案，且不會改變太小而顯然是僞檔案之檔案，因爲其架構或在檔案名稱中有既定內容。

明顯地，需要有改良後偵測病毒與其他類型惡意程式碼之技術。

本發明之觀點之一係提供一種辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在，該方法包括：起始化該電腦系統內之虛擬機台。該起始化後虛擬機台包括模擬一中央處理單元與記憶體之功能之軟體。該虛擬機台虛擬地執行標的程式，使得該標的程式只透過該虛擬機台來互動於該電腦系統。該方法包括：分析該標的程式在虛擬執行後之行爲以辨別惡意程式碼行爲之出現，並指示該惡意程式碼行爲之出現於一行爲圖樣內。結束該分析步驟後

五、發明說明（ψ）

係終止該虛擬機台，因而將包含於該虛擬機台內之該標的程式之複製從該電腦系統移除。

本發明之另一觀點係提供一種辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在。該方法包括：起始化該電腦系統內之一虛擬機台，該虛擬機台包括模擬一中央處理單元、記憶體與包括對該虛擬作業系統之中斷呼叫之一作業系統之功能之軟體。虛擬地執行該虛擬機台內之一標的程式，使得該標的程式只透過該虛擬機台來互動於該電腦系統。於虛擬執行期間，監測該標的程式之行爲以辨別惡意程式碼行爲之出現，並指示該惡意程式碼行爲之出現於一行爲圖樣內。終止該虛擬機台，留下該分析後標的程式之該行爲圖樣特徵之一記錄。

爲讓本發明之上述目的、特徵、和優點能更明顯易懂，下文特舉一較佳實施例，並配合所附圖式，作詳細說明如下：

圖式之簡單說明：

第 1 圖繪示依照分析行爲法所得之行爲圖樣，顯示未被電腦病程感染與被感染之程式碼之行爲圖樣。各位元係代表動作之旗標。位元之總串流係代表程式行爲之值。

第 2 圖繪示用於較佳分析偵測方法中之元件之方塊圖。

第 3 圖繪示 COM 檔案格式，當成程式架構拉出單元與程式載入單元之一例。

第 4 圖繪示虛擬 PC 至各程式檔案格式之介面。在開

五、發明說明（ 5 ）

始虛擬化之前，程式載入單元較好從程式檔案拉出正確項目點，程式碼與起始化資料。偏移至項目點程式碼之檔案係規定於程式表頭中，且其變化係有關於包含此程式之檔案類型。

第 5 圖顯示在載入二進位影像(.COM)程式與載入 MZ 可執行程式後之虛擬 PC 記憶體映對。爲以所需方法來虛擬化程式碼，虛擬 PC 與其記憶體映對之架構包括其可能之相同資訊，如同程式碼係執行於實體 PC 上，該實體 PC 係執行包含虛擬 PC 之虛擬機台。

第 6 圖提供顯示虛擬 PC 之較佳實施之元件。虛擬 PC 包含與實體 PC 之相同元件，除了所有虛擬 PC 元件係模擬於軟體，如執行於實體 PC 上之虛擬機台般。

較佳實施例

本發明之較佳實施例提供自動分析系統，其可偵測在電腦系統內之病毒與其他類型之惡意程式碼，藉由產生與接著分析導入至電腦系統之各電腦程式之行爲圖樣。新或改變後電腦程式係在電腦系統執行前被分析。較佳是，電腦系統起動代表電腦系統模擬之虛擬機台，且虛擬機台執行新的或改變後電腦程式以在新的電腦程式於實體電腦系統執行前，產生行爲圖樣。起始分析係執行於行爲圖樣上以辨認感染後程式，根據程式對電腦系統之起始代表。此分析系統也儲存行爲圖樣與相關之分析結果於資料庫內。新感染程式可由新產生行爲圖樣拉出所儲存之該程式之行爲圖樣，並分析所得之圖樣以辨認相關於惡意程式碼

五、發明說明（6）

之感染或圖樣。

各種不同名詞係用於程式化以描敘不同功能之程式子單元。在不同時間與不同程式語言，各種子單元係稱爲功能，程序，子程式，子程序與其他名稱。其所代表之此種設計與內文或差異對此討論並不明顯，所以此討論僅以程式來稱呼，指的是程式稱爲足夠在電腦系統或電腦環境內執行定義操作之任何大小之功能程式單元。此種特殊功能，如果在某些文書處理程式，比如包括微軟 word 文件之 VB(Visual Basic)巨集內由巨集所執行，係包括於此討論中。依此，各文件可視爲在此討論內文之程式。

爲方便與簡單，此討論參考已知名詞之病毒，如在受感染電腦系統內之自我傳播程式。在此所用，"視窗"是指由微軟公司在視窗品牌下所販賣之任何個人電腦作業系統。PC 或個人電腦係用於，在不管特殊改變下代表根據已知 X86 架構之電腦系統，包括由英特爾(INTEL)公司在奔騰品牌下所販賣之微處理器，與該微處理器與架構之後續者。此討論係用以描繪本發明之實施。本發明之觀點係找到在不同電腦系統之範圍內之應用程式，除了上述個人電腦系統外。

本發明人係分析各種不同病毒不其他惡意來源程式碼之行爲。某些一般病毒之特性係已辨認。病毒將感染其他程式，接著其他電腦以傳播。病毒包括感染回圈，其將病毒複製到另一可執行程式或有時在文件內，比如 VB 巨集病毒。病毒與特洛伊型病毒一般包含炸彈(payload)。此

五、發明說明（7）

炸彈允許病毒來響影被感染系統或傳播其存在。炸彈比如爲一種訊息，其彈跳出以宣稱該病毒或損壞受感染電腦之惡意功能，比如，由篡改或抹除硬碟上之資料，或改變或失能在 BIOS 快閃記憶體或 EEPROM 內之 BIOS。

另一種病毒之已知特徵是，病毒常駐於記憶體內。DOS 病毒必需複製其本身至記憶體內並常駐。大部份病毒並不使用明顯終端與常駐(TSR)呼叫，卻使用將病毒複製至高記憶體內之程序。病毒接著直接改變在高記憶體方塊內之資料。在此感染方式之另一觀點，中斷向量係被改變以指向已被記憶體常駐病毒或其他惡意程序所改變之記憶體方塊。這些改變後記憶體方塊係儲存感染程序。視窗病毒使其本身遇到 ring0，比如，使用呼叫閘或 DPMI 呼叫，並常駐於系統中。

這些行爲係病毒之特徵，且總體來說，並非爲不具惡意程式之特徵。因此，程式可辨認病毒或被病毒所感染，如果其處理某些這些行爲，這些行爲之某些集合或所有此種行爲。在本發明之實施例中，這些行爲或行爲總合之出現係由代表受感染程式之行爲特徵之行爲圖樣資料集內之位元集合所代表。正常與受感染檔案之行爲圖樣之例係顯示於第 1 圖中。

在本發明之較佳實施例中，新載入或呼叫程式之行爲係分析於虛擬機台中，虛擬機台係於軟體中模擬完整 PC 或足夠完整 PC，且其爲產生行爲(behavior)圖樣(pattern)之虛擬 PC。虛擬 PC 模擬新或改變後程式之執行，模擬系

五、發明說明（8）

統功能，且虛擬 PC 監測可能受感染程式之行爲，並記錄此行爲以分析來決定標的程式具病毒或惡意行爲。虛擬機台之虛擬執行結果係代表新程式之行爲圖樣。如底下所討論，由虛擬 PC 所產生之行爲圖樣係辨認出，程式係被病毒所感染或其本身爲病毒。使用虛擬執行與分析新程式之病毒之優點在於，虛擬機台是虛擬的，且如果虛擬化新程式包括病毒，只有此虛擬機台被感染。被感染之虛擬機台係在模擬後被刪除，使得此感染係不完整的，且病毒並不會傳播。行爲圖樣係在刪除虛擬機台後仍然留下來，允許分析程式來辨認病毒之存在與新程式內之感染。

較佳是，每次分析新程式時，虛擬機台之新例係產生，可由任何預先虛擬化程式 包括任何較早之分析過之病毒，所自由改變。新程式接著執行於新虛擬機台例上，較好是後續接著改變後中斷呼叫程序，其將於底下描敘。當虛擬機台執行新程式以及改變後中斷呼叫程序時，虛擬機台監測所有系統呼叫，DPMI/DOS 中斷與 I/O 埠讀/寫操作，根據所觀察之行爲而設定在行爲圖樣暫存器內之位元。行爲圖樣內之位元係保留於模擬完成後，且虛擬 PC 已終止。儲存於行爲圖樣暫存器內之位元係行爲圖樣，且代表是否虛擬執行程式包括代表病毒或其他惡意程式碼存在之行爲

改變後中斷呼叫程序係呼叫中斷，正被分析之程式係改變於虛擬 PC 內並產生這些中斷服務程序之各行爲圖樣。這允許本發明之較佳實施例來辨認只由中斷服務程序

五、發明說明（9）

所起始改變之某些類型病毒，且不開始傳播，直到改變後中斷係被另一程式所呼叫。藉由允許在虛擬機台內之各種中斷服務程序來被改變並接著分析改變後中斷，本發明之實施例係偵測此延遲傳播架構。

在某些較佳實施例中，只有分析行爲圖樣之統計與最後版本。這是有可能的，且在某些環境中是需要的，來監測行爲圖樣暫存器內之位元被設定之順序。行爲圖樣位元被設定之順序係提供額外資訊來允許辨認額外病毒行爲。追查行爲圖樣暫存器內位元之設定順序係完成於虛擬機台內。

分析行爲方法(ABM)之較佳實施係由改變後、新的、未知或可能被感染程式拉出行爲圖樣以及順序。行爲圖樣係較好用於分析未知程式之行爲以決定是否未知程式之行爲係惡意的。依此方法辨認之惡意行爲係允許載有檔案之病毒在感染主電腦系統前便能辨認出此病毒。行爲圖樣也可儲存於資料庫內，且虛擬機台可在改變後接著分析程式之行爲，以決定其功能係有可能被改變。這提供後感染分析。

上述之 ABM 係不同於傳統病毒偵測方法，其不同點在於其不將程式程式碼匹配於一組儲存圖樣，如簽名掃描系統與完整檢查系統所執行。甚至，虛擬機台係用以產生行爲圖樣與順序。所產生之行爲圖樣並不在版本更新間顯著改變，卻在當病毒感染程式時係顯著地改變。比如，當程式被取代或由新版程式所更新時，文書處理器將仍然類

五、發明說明（ㄩ）

似有文書處理器之行爲，但當文書處理器被病毒所感染時，其將顯著地改變。行爲圖樣所反映之差異係顯示於第1圖中。當文書處理器被檔案感染病毒所感染時，文書處理器將打開可執行檔案，並將病毒程式碼插入其中，因而感染其他檔案。在上述行爲圖樣中，這是完全被反映出來。

在本發明之較佳實施例中，分析程序特別係以感染方法爲目的，比如但不受限於將程式碼插入至其他可執行或文件內，將程式碼傳送至其他應用程式內以被傳送或儲存，將程式碼插入至高記憶體方塊內以及改變記憶體控制方塊。分析方法之較佳實施例更尋找破壞性內容，比如但不受限於刪除檔案或目錄。較佳是，此分析之例外，且不被辨認爲被感染，如程式之其他行爲特徵代表此程式係發展工具或軟體除錯工具，以及改變行爲係工具正常功能之一部份。發展工具之病毒感染可被偵測於特別功能非爲工具正常功能之一部份，也就是在發展過程中。皆存在於行爲圖樣內之主動(1)與被動(0)旗標係在此分析中非常明顯，連同過程發生順序。

根據本發明之較佳實施例，虛擬機台或虛擬 PC 代表完整電腦系統之模擬。完整電腦系統較好包括仿真 CPU，仿真記憶體，I/O 埠，BIOS 韌體，作業系統與作業系統資料。這相對於處理器之簡單仿真，其中只仿真處理器之動作。在仿真中，程式指令係由其原有格式轉換成執行相同功能於不同硬體平台上之一串指令。某些簽名掃描軟體係應用仿真以在可能被感染程式進行簽名掃描之前，分析可

五、發明說明（11）

能被感染程式之主動。在虛擬中，整個電腦係被仿真，包括作業系統呼叫，其未被真正執行但看起來呼叫程式係執行所需功能且回傳其被執行時之正確值。

如上述，虛擬 PC 包括 CPU，記憶體，I/O 埠，程式載入器與作業系統應用程式介面(API)項目點與介面。使用此完整虛擬 PC 係特別好，因爲其給予分析行爲方法對虛擬程式之高度控制，包括對作業系統 API 之複雜直接呼叫。虛擬化程式係未對實體機台之設備進行存取，因而避免可能之病毒或其他惡意程式碼由控制環境中脫離而感染主機系統之風險。

第 2 圖提供較好之分析行爲方法之整體架構，包括虛擬機台與主機系統元件間之關係。程式程式碼係藉由透過 I/O 埠位元複製而對硬碟直接存取來傳送至 ABM 引擎與分析系統，掛入(hook)至作業系統檔案或藉由依序掃描硬碟。程式碼係對已知檔案進行檢查。如果檔案係新的或被改變，其係被處理。所得之行爲簽名係被分析或比較，並儲存。病毒警告係回傳，當分析顯示此檔案包含惡意程式碼時。分析行爲方法較好包括：(1)檔案架構拉出；(2)改變偵測；(3)虛擬化；(4)分析與(5)決定。

在虛擬化程式之時，檔案格式，包括標的程式必需被評估。項目點程式碼係被拉出並載入至正確模擬偏移處之虛擬電腦記憶體。在實體電腦中，此功能可由爲作業系統之一部份之程式載入功能所執行。作業系統可執行保留於不同檔案格式之集合中之程式，比如

五、發明說明（7）

DOS 1.0 及/或 CP/M COM　二進位影像檔案，載於記憶體中之 100h，最大長度：64k。

DOS 2.0-DOS 7.1 EXE　MZ 型執行程式，表頭決定載入位址之 CS：IP。

Windows 3.0 執行程式　NE 型執行程式，其包含指向在 DOS 程式碼區之 DOS MZ 表頭與包括視窗(保護模式)程式碼之項目點之新可執行(NE)表頭。

OS/2 執行程式　LE/LX 型執行程式，其包含 DOS MZ-表頭與 DOS 程式碼區，以及由接續著 DOS 程式碼區段之 LE 表頭所決定之保護模式區。線性執行程式(LE)檔案係爲系統工具與裝置驅動程式而使用於視窗 3。LE 檔案係分區段。LX 檔案包含頁表儲存方式之差異，且用於 OS/2 作業系統。LE 檔案係分區段且區段係標號。

32 位元執行程式　PE 型執行程式，其包含 DOS MZ 表頭與 DOS 程式碼區，且可移動式執行程式表頭包含項目點與保護模式程式碼之檔案偏移。PE 檔案係分區段。

OLE 複合檔案　OLE 複合檔案(COM)係包含可執行格式串之文件檔案，通常稱爲巨集。所有辦公同元件包含應用之 VB，如同 IE 4 與 5 所包含。視窗 98 系統可由原始檔案執行 VB 程式碼。VB 程式碼係編輯並儲存於串流中，其係根據儲存於檔案表頭內之鏈結表列中之檔案偏移參考而分頁。

二進位影像　二進位影像係用於開機區段與主開機與分割表(master boot and partition table)。開機

五、發明說明（ㄐ）

區段與 MBR 包含在開始過程中載入於 0：7C00 記憶體處之可執行程式碼。

驅動程式檔案　　系統驅動程式係儲存成具表頭之二進位影像。此表頭包含關於儲存於檔案內之驅動程式。多重驅動程式可儲存於相同檔案內。

虛擬電腦載入功能係能處理檔案格式與上述之二進位影像檔案。載入功能係由虛擬化作業系統程式載入而執行，所以其根據使用於主機中之作業系統而改變。檔案架構分析過程係檢查檔案表頭與檔案架構以決定檔案格式，而非使用檔案延伸，因爲檔案延伸一般係不可靠的。上述之.EXE 格式因而包括 DLL，AX，OCX 與其他可執行檔案格式延伸。

複合文件檔案可包含如 VB 程式碼或巨集之可執行串流。複合文件檔案之架構係繪示於第 3 圖中。複合文件檔案之表頭包括鏈結表列(或檔案配置表)，其係參考於指向鏈結表列之項目點之目錄架構。在鏈結表列中之-1 值代表鏈之尾端。串流存在於方塊之外，其可依任何順序而散佈於檔案中之任何位置。在本發明之特殊實施例中，由複合文件檔案所拉出之程式碼係在導入至 VB 仿真器之前通過 VB 解編輯器。並非所有複合文件檔案包含 VB 程式碼。HTML 與 VBS 檔案可包含 VBS 程式碼當成文字。此程式碼係較好被拉出且當成在虛擬機台內之 VB 串流。

NE/PE/LE 執行檔案格式在複雜度上係相似，除非沒有使用鏈結表列，而這些檔案格式使用區段或頁表格。PE

五、發明說明（10）

檔案格式係根據 COFF 檔案格規。第 4 圖繪示根據本發明之某些實施例之如何這些檔案格式互動於較佳虛擬 PC。在仿真如何較佳虛擬 PC 互動於特殊檔案，檔案載入器較好決定如果存在之檔案係文件檔案或二進位檔案。

在檔案格式已仿真，且項目點檔案偏移已計算後，檔案係打開且虛擬機台讀取相關程式碼至記憶體內當成資料串流。程式碼長度係由檔案表頭之欄位內計算。此資訊係通過虛擬程式載入器。虛擬程式載入器使用於檔案表頭內之資訊來載入拉出程式碼於虛擬記憶體陣列內之正確模擬偏移。

記憶體映對工具係映對虛擬記憶體映對至被虛擬化之檔案類型之偏移：

DOS(CP/m)二進位影像檔案(.COM)　偏移 CS：100h

DOS(2.0 以上)執行格式檔案(MZ-EXE) 偏移 CS：從表頭之 IP

視窗 NE，PE，LE　偏移 C0000000+CS：從表頭之 IP

二進位影像 MBR，開機區段砠　偏移 0：7C00h

文件 COM 檔案，HTML 與 VBS 檔案　沒有特殊偏移，VBA 程式碼

每次虛擬化程式時，載入工具動態指定實體記憶體至虛擬電腦記憶體陣列，且接著建立新的虛擬機台。各虛擬機台包含 BIOS 資料區域，填滿之環境串區域，DOS 資料區域，記憶體控制方塊，程式區段字首區域，中斷向量表與描述符號表。虛擬機台之最後結構係取決於虛擬化程

五、發明說明（15）

式之類型。各虛擬化程式因而執行於更新記憶體區域，其係建立於當程式載入於虛擬 PC 時。先前例子，其中受感染程式係已虛擬化，因而無法影響後續程式之執行。當虛擬程式終止且虛擬機台完成標的虛擬化之行爲圖樣時，係關閉虛擬機台且釋出其記憶體資源。

第 5 圖繪示如果架構(COM)二進位影像檔案與 DOS 程式(MZ-EXE)檔案之虛擬記憶體。記憶體映對與映對工具係根據檔案類型而調整。

程式載入器係模擬作業系統之載入功能且建立代表在實體電腦內之相似系統區域之系統區域。這是特別有利的功能，因爲待評估之程式碼較好執行於如同其執行於實體電腦系統之方式。虛擬化程式係藉由從虛擬記憶體陣列擷取指令至預取指令佇列而執行。在此佇列中之指令係被解程式碼，其且長度係由其操作參數而決定。

指令指標係因此增加，使得指令載入器係備妥以擷取下一指令。虛擬機台係由指令操作之資料其上之操作參考之 r/m 欄位所決定。資料擷取機構係擷取此資料，並將此資料傳送邏輯單位，其接著執行由此程式碼所指定之操作。處理後資料之目的係由指令程式碼之參數而決定。資料寫入機制係用以將處理後資料寫入至仿真記憶體或仿真處理器暫存器組。此處理正確地反映發生於實體 CPU 內之操作。

所有處理之區域係被模擬，如第 6 圖所示。記憶體係 400KB 元件之陣列，所有記憶體存取係由記憶體映對

五、發明說明（16）

機構所映對。記憶體陣列之大小係可更進一步調整以適合於更大的程式。影像播放係由系統觀點來模擬，如映對於虛擬電腦記憶體映對中之 A000：0 與 BFFF：F(包括此)之記憶體之 128KB。標準 IMB PC 輸出入區域係由代表 I/O 埠 0-3FFh 之 1024 位元組之陣列所模擬。CPU 係藉由執行與實體 CPU 之相同低階功能來模擬於高階軟體中。

作業系統係實施於包含 BIOS 資料、DOS 資料區域，記憶體控制方塊與 DOS 裝置之欄位之 00h 位元組之記憶體陣列中之區域。中斷向量表佔據在記憶體陣列中之 1024(400h)位置，如同其在實體 PC 所佔據般。DOS 中斷結構係實施成回傳正確值之模擬功能，且藉由模擬 DOS 功能所預期之正確值而填滿記憶體陣列。

作業系統係實施成虛擬 API(VAPI)，其模擬由所有作業系統 API 所回傳之結果。

在虛擬化過程中，旗標係設定於行爲圖樣(Tstruct)欄位，當由這些欄位所代表之功能被虛擬化時。這些功能被呼叫之順序係記錄於排序器中。行爲圖樣因而使得待評估之程式行爲係相當符合於實體 PC 環境中之該程式之行爲。在執行虛擬化程式之過程中被改變之模擬中斷向量係在程式虛擬終止時被呼叫，因而當成於改變這些向量後，呼叫在實體電腦中之此種中斷向量之應用。

爲描繪此功能，考量可能執行於分析行爲法之操作中之下列操作組合：

尋找在此目錄中之第一 EXE 檔案　　；設定 FindFirst 旗

五、發明說明(17)

標(Tstruct 結構)

是否其爲 PE 可執行程式(檢查表頭)？ ；設定旗標 EXEcheck

如果不是，往前跳

否則：打開執行檔案 ；設定 EXEacc ess 旗標

寫入至區表 ；設定 EXEwrite 旗標

尋找檔案結束(EOF)；設定 EXEeof 旗標

寫至檔案 ；設定 EXEwrite 旗標

結束檔案 ；

尋找下一個 EXE 檔案 ；設定 FindNext 旗標

bit+1 64……1

回傳：0010 0100 1010 1001 0101 1101 1111 0010 1010 0010 0100 1001 0000 0101

值：24AA952F2A244905

排序器：21，22，23，24，26，29，8E，1，36，38，3B，3，9，C，F，13，16，1A，1C，1E，2B，2D，30，32，34

所得之行爲圖樣係 24AA952F2A244905

行爲圖樣包含代表使用者尚未有機會透過使用者輸入來互動於此操作之旗標(userInput 旗標未設定)。排序器包含位元設定之順序，其辨別上述之感染順序。因而，所觀察之行爲係最像病毒。

許多病毒係編程式碼，多樣態或使用"技巧"以避免被簽名掃描系統所偵測。無論使用何種技巧，行爲圖樣指向

五、發明說明 (18)

最顯著之病毒，因爲此種技巧係於一般應用中不會被用到。在任何情況中，本發明之較佳實施例需要，感染過程來呈現以觸發病毒警告以避免錯誤之正向警告。編程式碼後病毒係沒有問題的，因爲虛擬機台內之程式碼之執行，其產生行爲圖樣，係有效地將任何編程式碼後或多樣態病毒給予解程式碼，如同其在實體 PC 環境中一般。因爲虛擬電腦之所有部份係虛擬化於較佳實施例中，且虛擬程式決不允許來互動於實體電腦，病毒程式碼沒有機會從虛擬機台脫離且感染實體電腦。

變化偵測模組係以 6 階來比較現在檔案以決定如果檔案係先前分析：

- 檔案係相同(項目點程式碼，範例，檔案名稱與檔案大小係相同)。
- 檔案係不在資料庫中(新檔案)。
- 行爲圖樣符合於所儲存之圖樣。
- 檔案之項目程式碼係被改變。行爲圖樣係由先前儲存之圖樣中以二進位方式拉出。所得之位元圖樣係被分析。
- 檔案之項目程式碼，CRC 與表頭欄位係相同，但檔案係被重新命名。沒其他欄位被改變。
- 檔案之行爲圖樣係於資料庫中找到，且符合於已知之病毒行爲圖樣。
- 檔案之行爲圖樣係於資料庫中找到，且符合於已知之無害行爲圖樣。

如果檔案之可執行部份被改變的話，程式係被虛擬

五、發明說明（19）

化。不包含改變後可執行程式碼之檔案係不可能包含病毒，除非原始檔案係被感染。如果是這種情況，先前分析會偵測此病毒。當現有程式被更新時，其功能仍相同，因而其行為圖樣相當符合於其先前儲存之行為圖樣。如果改變後位元代表感染程序已加入，則檔案係視為被感染。

兩種偵測機制係一起操作，皆使用行為圖樣：

感染前偵測

這是最理想的情況。在感染前偵測中，行為圖樣係被分析，並發現能代表導入至系統之新或變化後程式之病毒行為。待評估之程式檔案可由移除病毒而修復或被刪除，如果病毒感染顯示太難而無法移除，或部份原始程式碼係被覆寫。感染後程式此時尚未執行於實體 PC 上，所以在發現病毒後，不需修復實體 PC。

感染後偵測

感染後偵測係發生於當起始感染係被感染前偵測所遺漏之情況。當病毒不執行任何病毒功能於第一次執行時並不改變指向感染程序之中斷向量時，此病毒可能被感染前偵測所遺漏。這是具稱為慢速感染與相似行為之惡意程式碼之情況。在感染後偵測，病毒在試著感染 PC 上之第一個可執行程式時，其會被偵測到。檔案掛入機制偵測對可執行程式(包括文件)之改變嘗試。ABM 引擎接著分析第一執行程式並發現其行為圖樣係改變成代表病毒在活動之方式。

資料庫架構

五、發明說明（ 20 ）

檔案 ID 區：　行爲圖樣，程式名稱，檔案大小與路徑。

修復結構　表頭欄位，區表與位置變化表。

區段表　在區表中之各區之大小與偏移(只用於視窗程式)

在文件中之巨集病毒係被視爲其爲執行程式。原始 VB 程式碼係被分析與逆編輯 VB 文件(COM)串流所復原(可應用之處)。所得之來源程式碼係不被儲存，也不被顯示來保護合法 VB 軟體之原始發表者。在虛擬化後，來源程式碼係被丟棄。

上述之病毒偵測系統之缺點是，起始分析係慢於圖樣掃描。此缺點係更甚於系統優點之偏移。使用檔案系統掛入代表所有新的檔案係被報導並分析於背景中。這代表，一旦電腦係免於病毒，完整掃描一般不再需要，除非保護系統在安裝新程式時被關閉。在簽名掃描式保護系統中，每次病毒簽名資料庫被更新時，電腦需要完整地再掃描。當使用者起動後續磁碟掃描時，未改變之檔案不需再次虛擬化，使得此動作係至少與圖樣掃描一樣快，但保護性更高。所儲存之資訊也有助於修復檔案或系統區之病毒破壞，在大多情況下，完整保護或有效地完整復原。

在測試 ABM 系統之芻型實施中，感染前偵測(96%)與感染後偵測(4%)之組合導致 100%偵測到所有已知病毒，利用新、改變後與已知病毒之組合。其他方法只偵測到 100%之已知病毒，並無法偵測到新、改變後或未知病毒。無法引用關於簽名掃描式產品之測試正確圖。此種產

五、發明說明（7）

品之結果係已知、改變後、新、未知病毒之混合之直接代表，比如，如果病毒測試組合之 30%係改變後、新、未知，則反映出之最後結果係接近於 30%遺漏病毒。本系統之較佳觀點之實施中，並沒有此種關係存在，而偵測率並不適當隨著現在病毒混合之改變而變化。

綜上所述，雖然本發明已以一較佳實施例揭露如上，然其並非用以限定本發明，任何熟習此技藝者，在不脫離本發明之精神和範圍內，當可作各種之更動與潤飾，因此本發明之保護範圍當視後附之申請專利範圍所界定者爲準。

四、中文發明摘要（發明之名稱：電腦免疫系統及在電腦系統中偵測不必要程式碼之方法）

一種自動分析系統，藉由產生與接續分析導入至該電腦系統之各電腦程式之行爲圖樣而偵測電腦系統內之惡意程式碼。該行爲圖樣之產生係由該電腦系統內之一虛擬機台來完成。起始分析可執行於該行爲圖樣上以辨別該電腦系統之該程式之起始化上之感染後程式。分析系統也儲存行爲圖樣與其相關分析結果在資料庫中之順序。新感染之程式可由參考所儲存之行爲圖樣而分析新產生行爲圖樣來偵測，以辨別感染或炸彈圖樣之存在。

英文發明摘要（發明之名稱：）

六、申請專利範圍

1. 一種辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，該方法包括：

起始化該電腦系統內之一虛擬機台，該虛擬機台包括模擬一中央處理單元與記憶體之功能之軟體；

虛擬地執行該虛擬機台內之一標的程式，使得該標的程式只透過該虛擬機台來互動於該電腦系統；

分析該標的程式在虛擬執行後之行爲以辨別惡意程式碼行爲之出現，並指示該惡意程式碼行爲之出現於一行爲圖樣內；以及

在該分析步驟後終止該虛擬機台，因而將包含於該虛擬機台內之該標的程式之複製從該電腦系統移除。

2.如申請專利範圍第 1 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該虛擬機台模擬輸出入埠、作業系統資料區與一作業系統應用程式介面之功能。

3.如申請專利範圍第 2 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該虛擬機台更包括一虛擬 VB(Visual Basic)引擎。

4.如申請專利範圍第 2 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該標的程式之虛擬執行造成該標的程式來互動於該模擬後作業系統應用程式介面。

5.如申請專利範圍第 1 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該標的程式

六、申請專利範圍

係新導入至該電腦系統，並在虛擬執行該標的程式之前不會被執行。

6.如申請專利範圍第 1 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中在一第一程式之一第一例被該虛擬機台所分析且產生一第一行爲圖樣並儲存於該電腦系統內之一資料庫之後，該方法更包括：

決定該第一程式係被改變；

藉由執行該虛擬機台內之該改變後第一程式來分析該改變後第一程式，以提供一第二行爲圖樣；以及

比較該第一行爲圖樣與該第二行爲圖樣。

7.如申請專利範圍第 6 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中每次該第一程式被改變時，係產生一新行爲圖樣。

8.如申請專利範圍第 6 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中在改變該第一程式期間之惡意程式碼之導入係藉由比較該第一行爲圖樣與該第二行爲圖樣而偵測。

9.如申請專利範圍第 6 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中當該改變後第一程式係該第一程式之新版本時，該第一行爲圖樣係本質上相似於該第二行爲圖樣。

10.如申請專利範圍第 1 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該行爲圖樣係辨認執行於該標的程式之該虛擬執行內之功能，該方法

六、申請專利範圍

更包括追蹤該功能被該虛擬機台內之該標的程式虛擬執行之順序。

11. 一種辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，該方法包括：

起始化該電腦系統內之一虛擬機台，該虛擬機台包括模擬一中央處理單元、記憶體與包括對該虛擬作業系統之中斷呼叫之一作業系統之功能之軟體；

虛擬地執行該虛擬機台內之一標的程式，使得該標的程式只透過該虛擬機台來互動於該電腦系統；

監測虛擬執行期間之該標的程式之行爲以辨別惡意程式碼行爲之出現，並指示該惡意程式碼行爲之出現於一行爲圖樣內；以及

終止該虛擬機台，留下該分析後標的程式之該行爲圖樣特徵之一記錄。

12.如申請專利範圍第 11 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該記錄係在該電腦系統內之一行爲暫存器中。

13.如申請專利範圍第 11 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中在一第一程式之一第一例被該虛擬機台所分析且產生一第一行爲圖樣並儲存於該電腦系統內之一資料庫之後，該方法更包括：

決定該第一程式係被改變；

藉由執行該虛擬機台內之該改變後第一程式來分析

六、申請專利範圍

該改變後第一程式，以提供一第二行爲圖樣；以及

比較該第一行爲圖樣與該第二行爲圖樣。

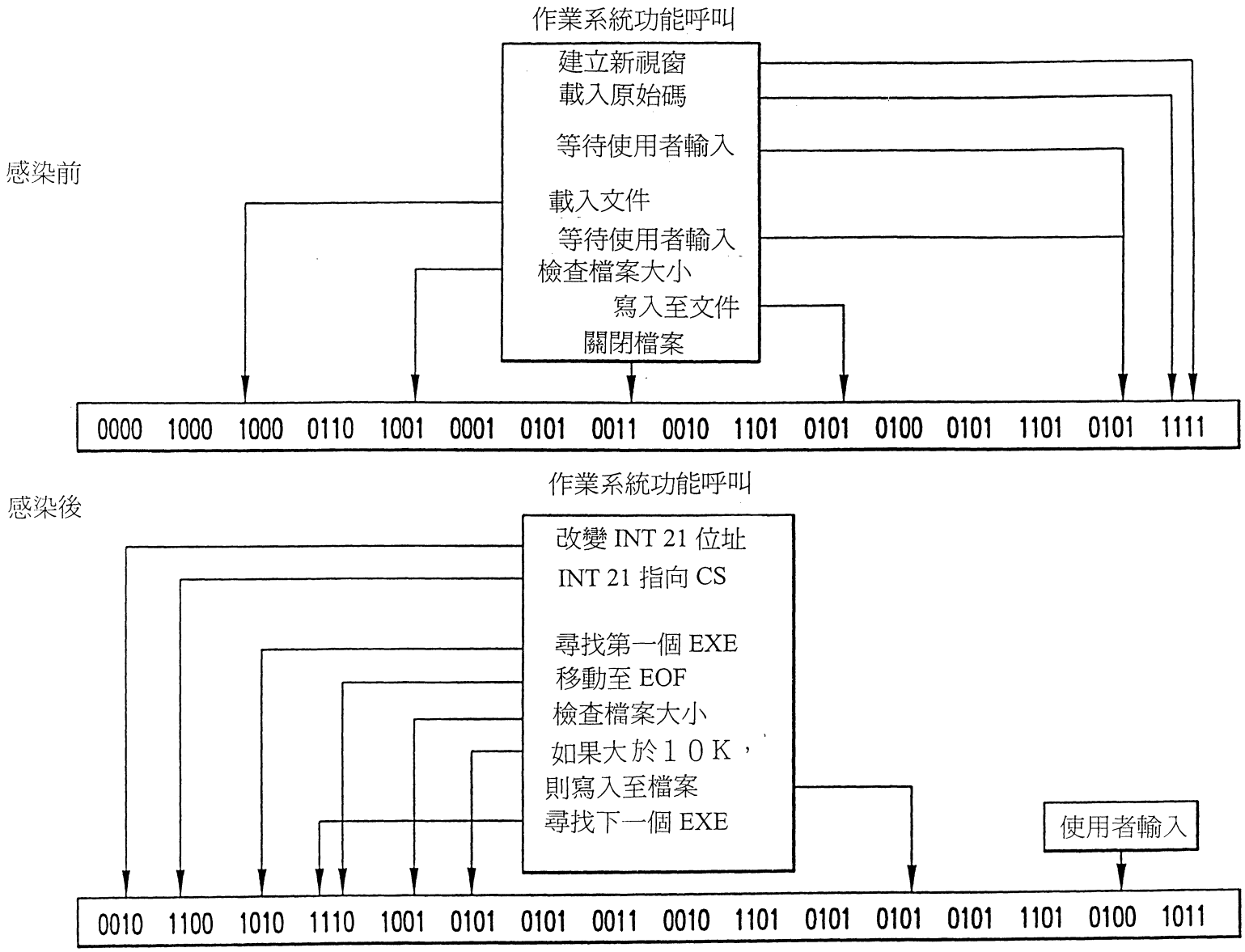
14.如申請專利範圍第 13 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中每次該第一程式被改變時，係產生一新行爲圖樣。

15.如申請專利範圍第 13 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中在改變該第一程式期間之惡意程式碼之導入係藉由比較該第一行爲圖樣與該第二行爲圖樣而偵測。

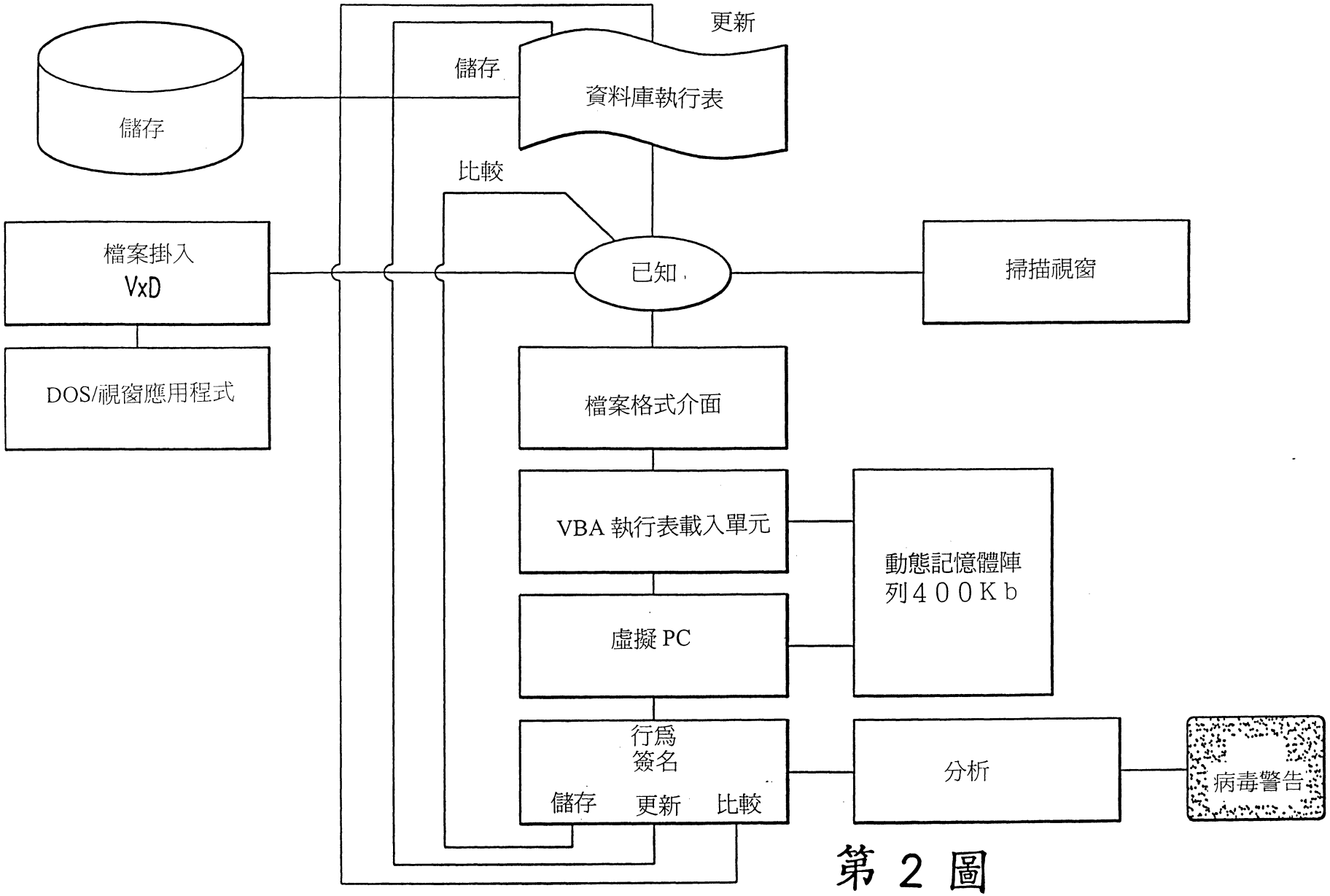
16.如申請專利範圍第 13 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中當該改變後第一程式係該第一程式之新版本時，該第一行爲圖樣係本質上相似於該第二行爲圖樣。

17.如申請專利範圍第 13 項所述之辨別在一電腦系統內之程式程式碼內之惡意程式碼之存在的方法，其中該行爲圖樣係辨認執行於該標的程式之該虛擬執行內之功能，該方法更包括追蹤該功能被該虛擬機台內之該標的程式虛擬執行之順序。

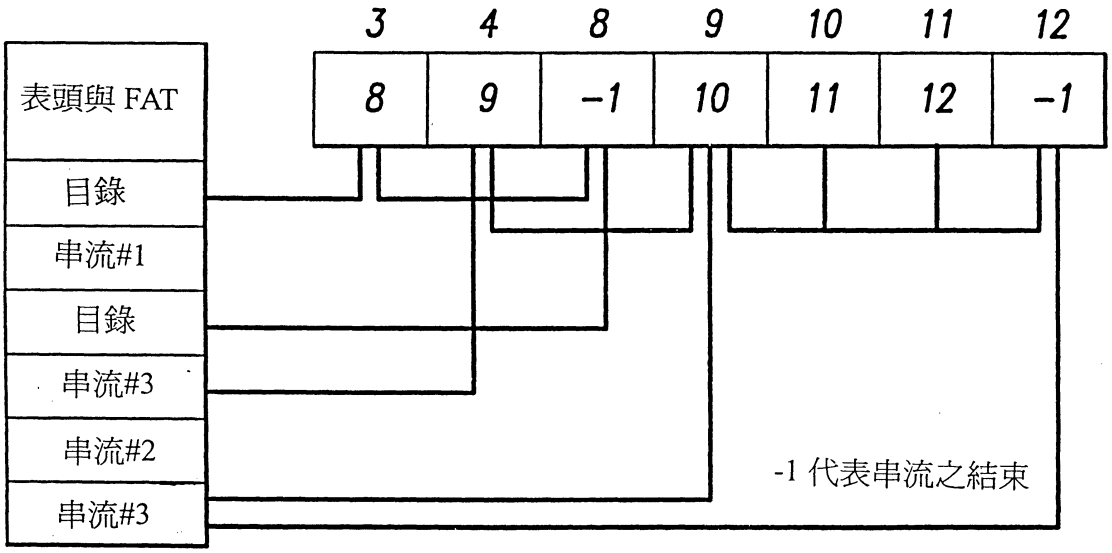
被[illegible]案感染病毒所感染前後之行爲圖樣



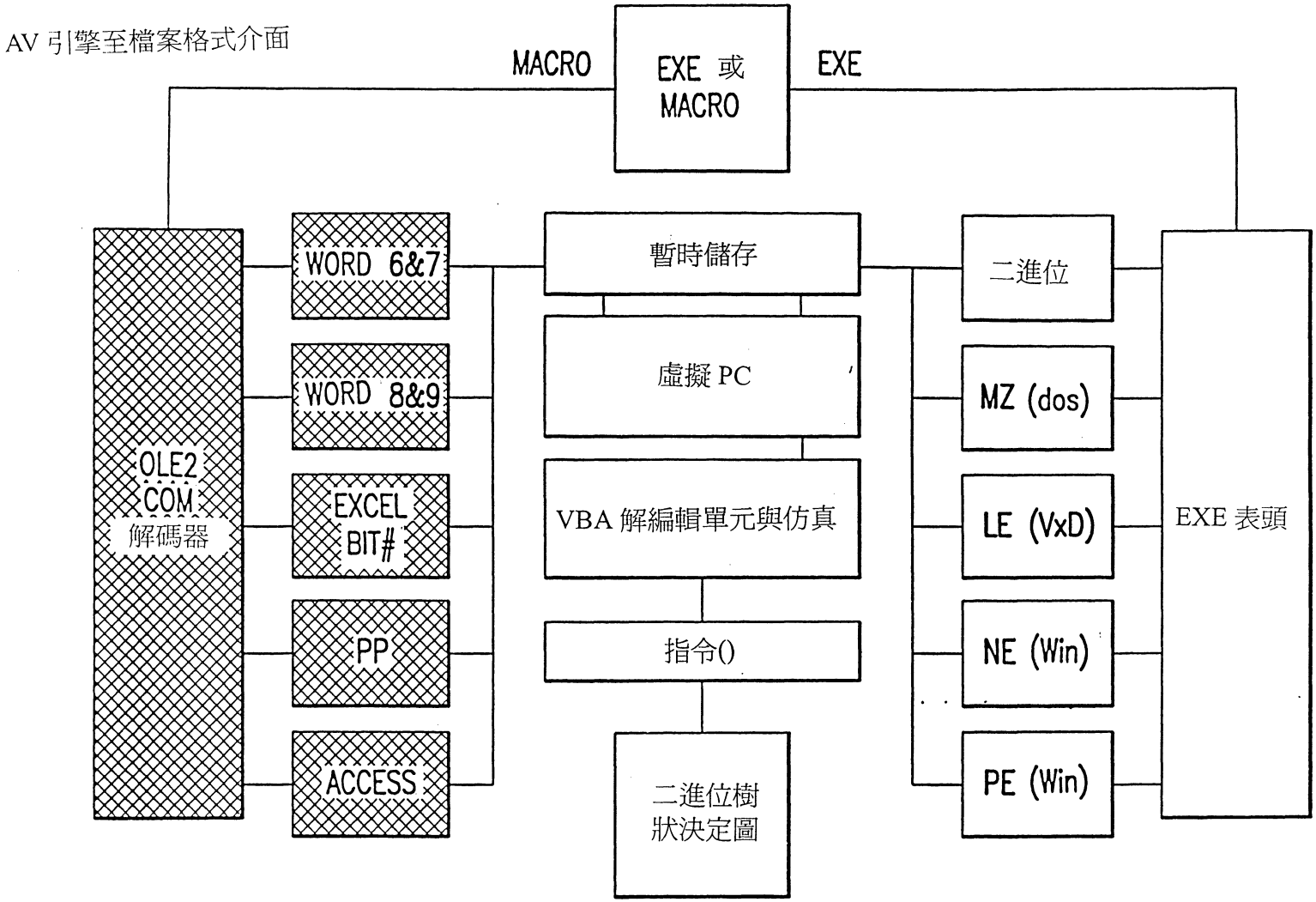
第 1 圖


更新
儲存
儲存
資料庫執行表
比較
檔案掛入
VxD
已知
掃描視窗
DOS/視窗應用程式
檔案格式介面
VBA 執行表載入單元
動態記憶體陣
列４００Ｋｂ
虛擬 PC
行爲
簽名
儲存
更新
比較
分析
病毒警告

第 2 圖


3
4
8
9
10
11
12
表頭與 FAT
8
9
-1
10
11
12
-1
目錄
串流#1
目錄
串流#3
串流#2
串流#3
-1 代表串流之結束

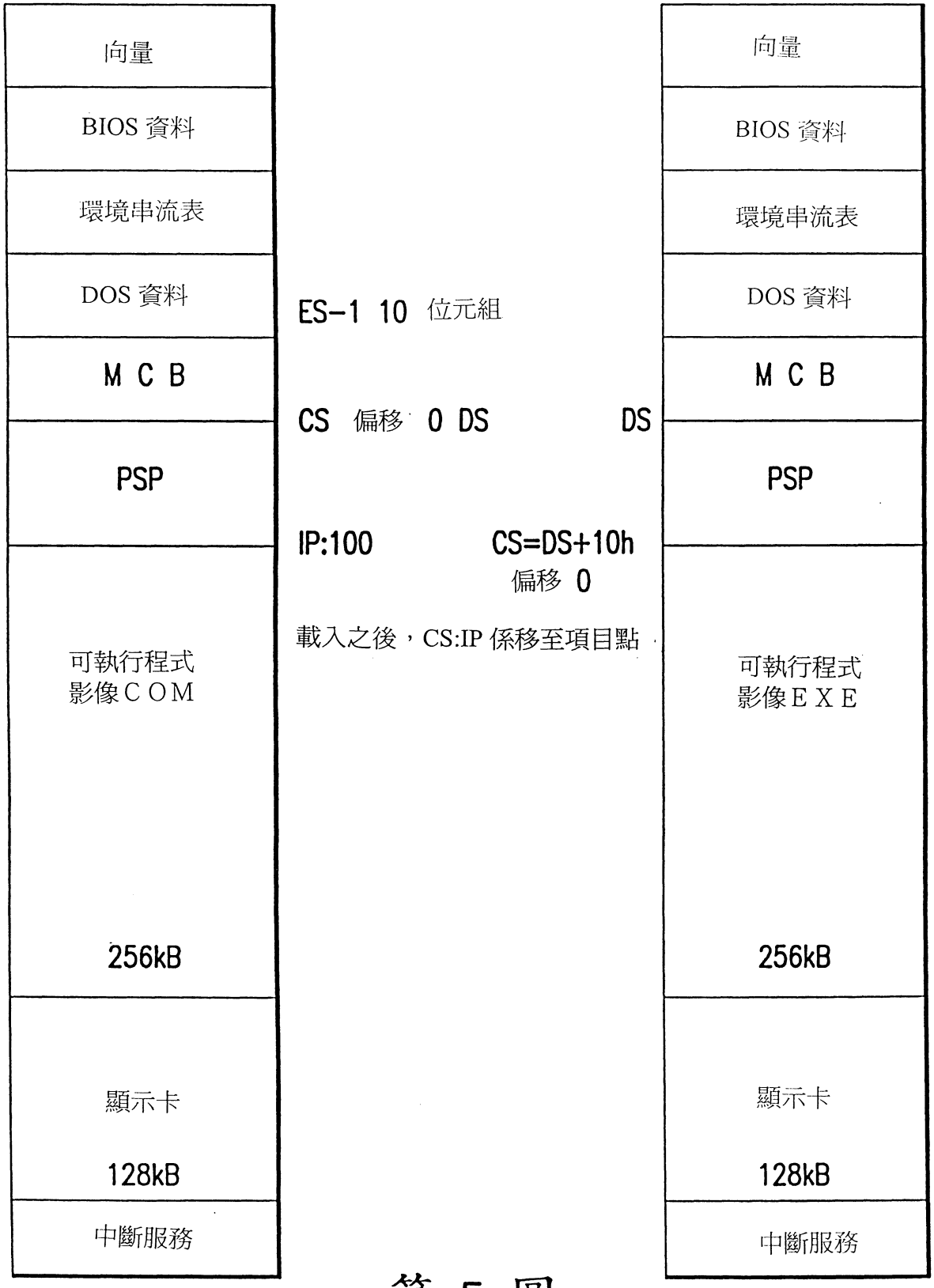
第 3 圖


AV 引擎至檔案格式介面
MACRO
EXE 或
MACRO
EXE
OLE2
COM
解碼器
WORD 6&7
WORD 8&9
EXCEL
BIT#
PP
ACCESS
暫時儲存
虛擬 PC
VBA 解編輯單元與仿真
指令()
二進位樹
狀決定圖
二進位
MZ (dos)
LE (VxD)
NE (Win)
PE (Win)
EXE 表頭

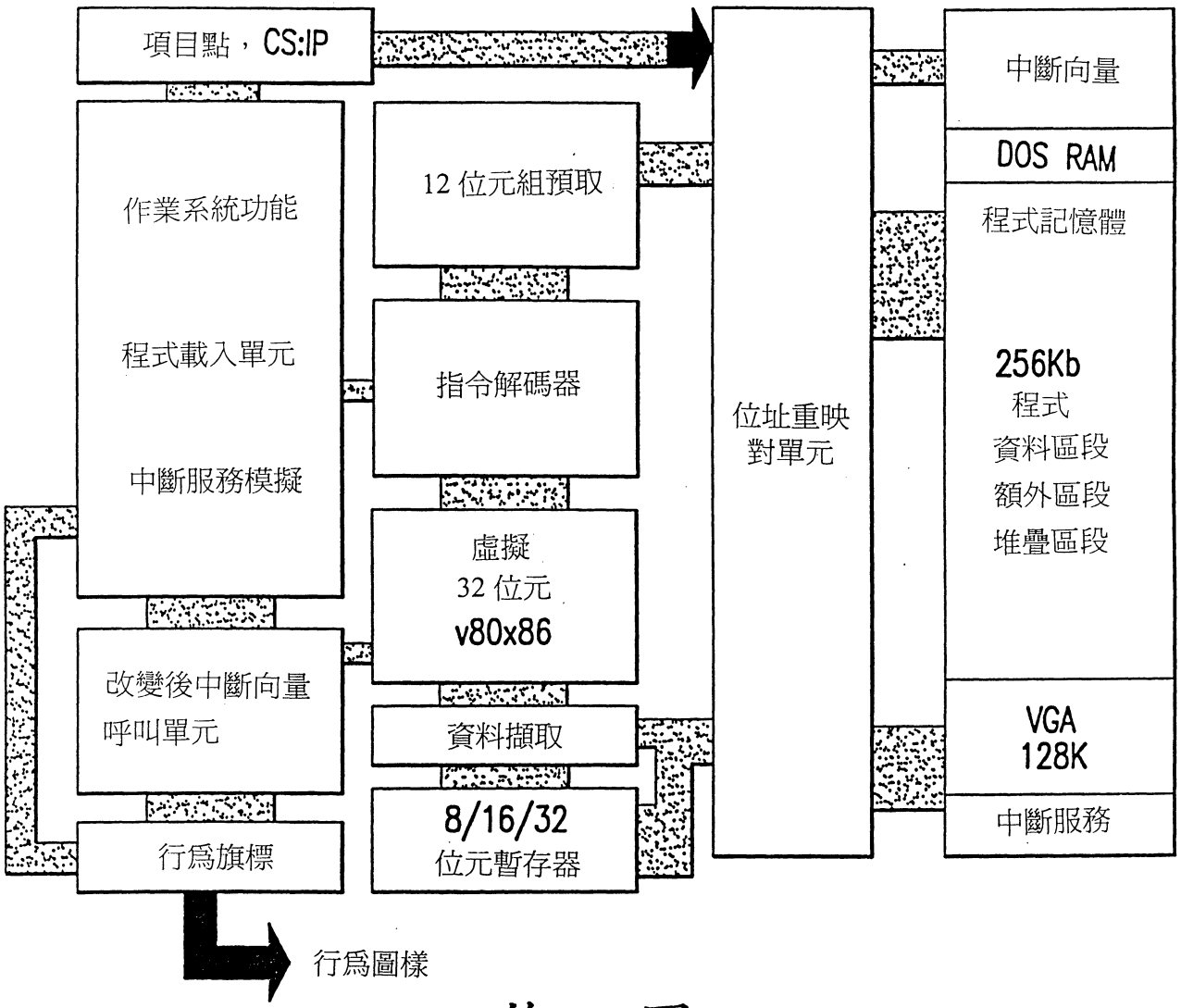
第 4 圖

V80X86

二進位 COM 與 EXE 檔案之記憶體映對



第 5 圖


項目點，CS:IP
作業系統功能
程式載入單元
中斷服務模擬
改變後中斷向量
呼叫單元
行爲旗標
12 位元組預取
指令解碼器
虛擬
32 位元
v80x86
資料擷取
8/16/32
位元暫存器
位址重映
對單元
中斷向量
DOS RAM
程式記憶體
256Kb
程式
資料區段
額外區段
堆疊區段
VGA
128K
中斷服務
行爲圖樣

第 6 圖