



(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 201328363 A1

(43) 公開日：中華民國 102 (2013) 年 07 月 01 日

(21) 申請案號：101140398

(22) 申請日：中華民國 101 (2012) 年 10 月 31 日

(51) Int. Cl. : *H04N7/26 (2006.01)*

(30) 優先權：2011/11/01 歐洲專利局 11187410.3

(71) 申請人：進益研究公司 (加拿大) RESEARCH IN MOTION LIMITED (CA)
加拿大

(72) 發明人：恩古言 恩古言 NGUYEN, NGUYEN (CA) ; 計天穎 JI, TIANYING (US) ; 何大可 HE, DAKE (CA)

(74) 代理人：陳長文

申請實體審查：有 申請專利範圍項數：20 項 圖式數：13 共 61 頁

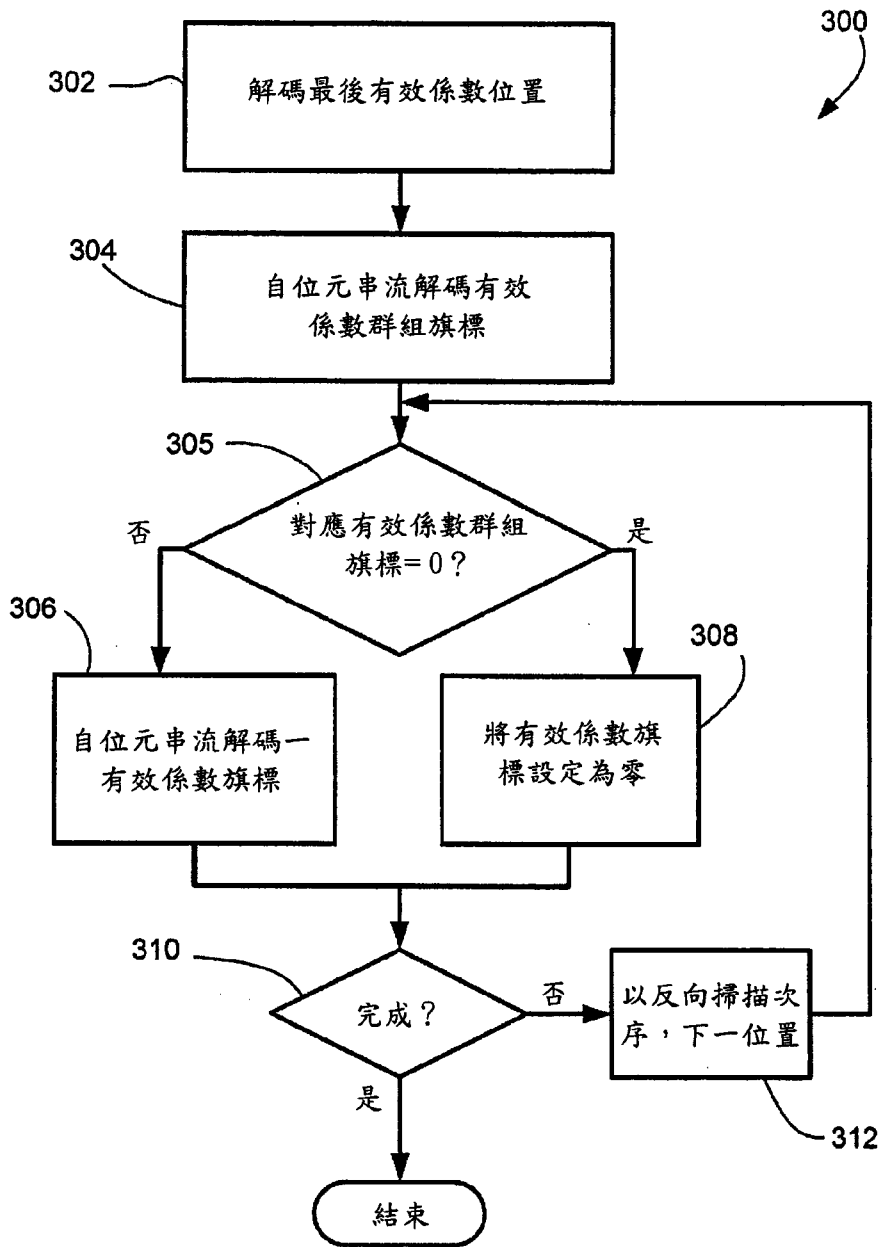
(54) 名稱

用於編碼及解碼之多層級有效性映射

MULTI-LEVEL SIGNIFICANCE MAPS FOR ENCODING AND DECODING

(57) 摘要

本發明闡述用於視訊資料之編碼及解碼之方法，其中在編碼及解碼程序中使用多層級有效性映射。將形成該有效性映射之有效係數旗標分組成若干連續群組，且一有效係數群組旗標針對每一群組表示彼群組是否不含有非零有效係數旗標。若在該群組中不存在非零有效係數旗標，則將該有效係數群組旗標設定為零。將有效係數群組旗標集合編碼成位元串流。將歸屬於具有係非零之一有效係數群組旗標之一群組內之任何有效係數旗標編碼成該位元串流，而不將歸屬於具有係零之一有效係數群組旗標之一群組內之有效係數旗標編碼成該位元串流。





(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 201328363 A1

(43)公開日：中華民國 102 (2013) 年 07 月 01 日

(21)申請案號：101140398

(22)申請日：中華民國 101 (2012) 年 10 月 31 日

(51)Int. Cl. : *H04N7/26 (2006.01)*

(30)優先權：2011/11/01 歐洲專利局 11187410.3

(71)申請人：進益研究公司 (加拿大) RESEARCH IN MOTION LIMITED (CA)
加拿大

(72)發明人：恩古言 恩古言 NGUYEN, NGUYEN (CA) ; 計天穎 JI, TIANYING (US) ; 何大可 HE, DAKE (CA)

(74)代理人：陳長文

申請實體審查：有 申請專利範圍項數：20 項 圖式數：13 共 61 頁

(54)名稱

用於編碼及解碼之多層級有效性映射

MULTI-LEVEL SIGNIFICANCE MAPS FOR ENCODING AND DECODING

(57)摘要

本發明闡述用於視訊資料之編碼及解碼之方法，其中在編碼及解碼程序中使用多層級有效性映射。將形成該有效性映射之有效係數旗標分組成若干連續群組，且一有效係數群組旗標針對每一群組表示彼群組是否不含有非零有效係數旗標。若在該群組中不存在非零有效係數旗標，則將該有效係數群組旗標設定為零。將有效係數群組旗標集合編碼成位元串流。將歸屬於具有係非零之一有效係數群組旗標之一群組內之任何有效係數旗標編碼成該位元串流，而不將歸屬於具有係零之一有效係數群組旗標之一群組內之有效係數旗標編碼成該位元串流。

發明專利說明書

102年2月6日修正

(本說明書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫)

※申請案號：101140398

※申請日：101.10.31

※IPC分類：H04N7/26(2006.01)

一、發明名稱：(中文/英文)

用於編碼及解碼之多層級有效性映射

MULTI-LEVEL SIGNIFICANCE MAPS FOR ENCODING AND
DECODING

二、中文發明摘要：

本發明闡述用於視訊資料之編碼及解碼之方法，其中在編碼及解碼程序中使用多層級有效性映射。將形成該有效性映射之有效係數旗標分組成若干連續群組，且一有效係數群組旗標針對每一群組表示彼群組是否不含有非零有效係數旗標。若在該群組中不存在非零有效係數旗標，則將該有效係數群組旗標設定為零。將有效係數群組旗標集合編碼成位元串流。將歸屬於具有係非零之一有效係數群組旗標之一群組內之任何有效係數旗標編碼成該位元串流，而不將歸屬於具有係零之一有效係數群組旗標之一群組內之有效係數旗標編碼成該位元串流。

三、英文發明摘要：

Methods of encoding and decoding for video data are described in which multi-level significance maps are used in the encoding and decoding processes. The significant-coefficient flags that form the significance map are grouped into contiguous groups, and a significant-coefficient-group flag signifies for each group whether that group contains no non-zero significant-coefficient flags. If there are no non-zero significant-coefficient flags in the group, then the significant-coefficient-group flag is set to zero. The set of significant-coefficient-group flags is encoded in the bitstream. Any significant-coefficient flags that fall within a group that has a significant-coefficient-group flag that is non-zero are encoded in the bitstream, whereas significant-coefficient flags that fall within a group that has a significant-coefficient-group flag that is zero are not encoded in the bitstream.

四、指定代表圖：

(一)本案指定代表圖為：第(7)圖。

(二)本代表圖之元件符號簡單說明：

(無元件符號說明)

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

(無)

六、發明說明：

【發明所屬之技術領域】

本申請案一般而言係關於資料壓縮，且特定而言係關於用於使用多層級有效性映射編碼及解碼視訊之方法及裝置。

本文件之揭示內容及隨附材料之一部分含有對其主張版權所有之材料。如同專利與商標局檔案或記錄中所顯現，版權所有者不反對任何人對本專利文件或本專利揭示內容進行拓製，但另外保留所有其他版權。

【先前技術】

資料壓縮發生於若干個內容脈絡中。資料壓縮極常用於通信及電腦網路連線中以高效地儲存、傳輸及重現資訊。資料壓縮在影像、音訊及視訊之編碼中具有特定應用。視訊由於每一視訊圖框所需之大量資料以及編碼及解碼通常需要發生之速度而對資料壓縮提出一重要挑戰。用於視訊編碼之當前最佳技術係ITU-T H.264/AVC視訊編碼標準。其針對不同應用定義若干個不同設定檔，包含主要設定檔、基線設定檔及其他。當前正透過MPEG-ITU之一聯合倡議開發下一代視訊編碼標準：高效視訊編碼(HEVC)。

存在用於編碼/解碼影像及視訊之若干個標準，包含使用基於區塊之編碼程序之H.264。在此等程序中，將影像或圖框劃分成若干區塊(通常4×4或8×8)，且將該等區塊光譜地變換成經量化且經熵編碼之係數。在諸多情形中，經

變換之資料並非實際像素資料，而是在一預測操作之後的殘餘資料。預測可係圖框內的(亦即，在圖框/影像內之區塊至區塊)或圖框間的(亦即，在圖框之間)(亦稱為運動預測)。預期HEVC(亦稱為H.265)將亦具有此等特徵。

當光譜地變換殘餘資料時，此等標準中之諸多標準規定使用一離散餘弦變換(DCT)或對其之某一變化形式。然後使用一量化器量化所得DCT係數以產生經量化變換域係數或索引。

然後使用一特定內容脈絡模型將經量化變換域係數之區塊或矩陣(有時稱作一「變換單元」)熵編碼。在H.264/AVC及HEVC之當前開發工作中，藉由以下操作編碼經量化變換係數：(a)編碼指示區塊中之最後非零係數之位置之一最後有效係數位置；(b)編碼指示含有非零係數的區塊中之位置(除最後有效係數位置之外)之一有效性映射；(c)編碼非零係數之量值；及(d)編碼非零係數之正負號。經量化變換係數之此編碼通常佔據位元串流中之經編碼資料之30%至80%。

變換單元通常係 $N \times N$ 。常見大小包含 4×4 、 8×8 、 16×16 及 32×32 ，但其他大小係可能的。有效性映射中之符號之熵編碼係基於一內容脈絡模型。在一 4×4 明度或色度區塊或變換單元(TU)之情形中，一單獨內容脈絡係與TU中之每一係數位置相關聯。亦即，編碼器及解碼器針對 4×4 明度及色度TU追蹤總共30個(不包含右底部角位置)單獨內容脈絡。將 8×8 TU分割(在概念上出於內容脈絡相關聯之目

的)成 2×2 區塊以使得一個相異內容脈絡與 8×8 TU中之每一 2×2 區塊相關聯。因此，編碼器及解碼器針對 8×8 明度及色度TU追蹤總共 $16+16=32$ 個內容脈絡。此意指在有效性映射之編碼及解碼期間編碼器及解碼器追蹤且查找62個不同內容脈絡。當計及 16×16 TU及 32×32 TU時，所涉及之相異內容脈絡之總數目係88。在額外26個內容脈絡當中，13個內容脈絡用於明度TU且13個內容脈絡用於色度TU。13個內容脈絡至 16×16 或 32×32 TU中之係數位置之指派如下。令 (r, c) 表示TU中之一位置，其中若TU係為 16×16 之大小則 $0\leq r, c\leq 15$ ，且若TU係為 32×32 之大小則 $0\leq r, c\leq 31$ 。然後將3個相異內容脈絡指派至左上角處之三個位置 $(0, 0)$ 、 $(0, 1)$ 、 $(1, 0)$ ，包含DC位置 $(0, 0)$ ；將5個相異內容脈絡指派至區域 $\{(r, c): 2\leq r+c<5\}$ 中之位置；且將最後5個相異內容脈絡指派至所有剩餘位置。除 $(0, 0)$ 、 $(0, 10)$ 及 $(1, 0)$ 之前3個內容脈絡之外，區域 $\{(r, c): 2\leq r+c<5\}$ 中之一位置之內容脈絡之推導取決於其右下部鄰近者。令 $s(r, c)$ 表示位置 (r, c) 處之一係數之有效旗標，亦即，若該係數不為零則 $s(r, c)=1$ ，且否則 $s(r, c)=0$ 。位置 (r, c) 之內容脈絡等於 $\min(s(r+1, c)+s(r, c+1)+s(r+2, c)+s(r, c+2)+s(r+1, c+1), 4)$ ，其中 $\min(a, b)$ 傳回 a 與 b 之間的較小值。類似地推導剩餘區域 $\{(r, c): r+c\geq 5\}$ 中之一位置 (r, c) 之內容脈絡。

4×4 及 8×8 有效性映射之內容脈絡係藉由位元位置而判定。 16×16 及 32×32 有效性映射之內容脈絡主要係藉由相鄰位元之值而判定。 16×16 及 32×32 有效性映射之內容脈絡之

判定需要相當大量計算，此乃因在多數情形中處理器藉由查看相鄰有效旗標之值來判定內容脈絡，此涉及高成本之記憶體存取操作。

【發明內容】

本申請案闡述用於藉助內容脈絡自適應編碼或解碼來編碼及解碼有效性映射之方法及編碼器/解碼器。編碼器及解碼器使用多層級有效性映射。在至少一種情形中，與較大變換單元(諸如， 16×16 及 32×32 TU)一起使用多層級映射。

在一項態樣中，本申請案闡述一種自經編碼資料之一位元串流重新建構一變換單元之有效係數旗標之方法。該方法包含：重新建構有效係數群組旗標，其中每一有效係數群組旗標對應於一各別有效係數旗標群組；及藉由以下操作來重新建構每一有效係數旗標，若該有效係數旗標在具有係非零之對應有效係數群組旗標之一群組中，則自該位元串流解碼彼有效係數旗標，且若該有效係數旗標在具有係零之對應有效係數群組旗標之一群組中，則將彼有效係數旗標設定為零。

在另一態樣中，本申請案闡述一種用於編碼一變換單元之有效係數旗標之方法。該方法包含：編碼有效係數群組旗標，其中每一有效係數群組旗標對應於一各別有效係數旗標群組，且其中將該有效係數群組旗標設定為零以指示彼對應群組不含有非零有效係數旗標；及針對該等有效係數旗標中之每一者，若該有效係數旗標在其對應有效係數

群組旗標係非零所針對的群組中之一者中，則編碼彼有效係數旗標，且若該有效係數旗標在該對應有效係數群組旗標係零所針對的群組中之一者中，則不編碼彼有效係數旗標。

在一項態樣中，以一規定次序(諸如，一正向或反向掃描次序)執行該等有效係數旗標之重新建構。在另一態樣中，以一規定次序執行該等有效係數群組旗標之重新建構，該規定次序可係與重新建構該等有效係數旗標所使用相同之規定次序，諸如，正向或反向掃描次序。

在一項態樣中，若對應群組中之所有有效係數旗標皆係零，則將一有效係數群組旗標設定為零。在另一態樣中，若對應群組中之至少一個有效係數旗標係非零，則一有效係數群組旗標係非零。在另一態樣中，甚至在對應群組中之所有有效係數旗標皆係零之情況下，一特殊情形亦可產生係非零之一有效係數群組旗標，在該情形中，解碼器將自位元串流解碼彼對應群組之所有零值有效係數旗標。

在再一態樣中，本申請案闡述一種自經編碼資料之一位元串流重新建構一變換單元之有效係數旗標之方法。該方法包含：自該位元串流解碼有效係數群組旗標，其中每一有效係數群組旗標對應於一各別有效係數旗標群組，且其中若彼對應群組不含有非零有效係數旗標，則每一有效係數群組旗標係零；及針對每一有效係數群組旗標，若彼有效係數群組旗標係非零，則自位元串流解碼彼對應有效係數旗標群組之有效係數旗標，且否則，將彼對應有效係數

旗標群組中之所有有效係數旗標設定為零。

在再一態樣中，本申請案闡述一種自經編碼資料之一位元串流重新建構一變換單元之有效係數旗標之方法，其中該變換單元被分割成若干非重疊有效係數旗標群組。該方法包含：藉由將至少一個有效係數群組旗標之值設定為1來重新建構有效係數群組旗標，其中每一有效係數群組旗標對應於一各別有效係數旗標群組；及藉由以下操作來重新建構每一有效係數旗標：若該有效係數旗標在具有係非零之對應有效係數群組旗標之一群組中，則自該位元串流解碼彼有效係數旗標，且若該有效係數旗標在具有係零之對應有效係數群組旗標之一群組中，則將彼有效係數旗標設定為零。

在又一態樣中，本申請案闡述一種用於編碼一變換單元之有效係數旗標之方法，其中將該變換單元分割成若干非重疊有效係數旗標群組。該方法包含：編碼有效係數群組旗標，其中每一有效係數群組旗標對應於有效係數旗標群組中之一各別者，且其中將該有效係數群組旗標設定為零以指示彼對應群組不含有非零有效係數旗標，且其中該編碼不包含經推斷為1之至少一個有效係數群組旗標而不論其對應群組是否含有非零有效係數旗標。該方法進一步包含，針對該等有效係數旗標中之每一者，若該有效係數旗標在其對應有效係數群組旗標係非零所針對的群組中之一者中，則編碼彼有效係數旗標，且若該有效係數旗標在該對應有效係數群組旗標係零所針對的群組中之一者中，則

不編碼彼有效係數旗標。

在又一態樣中，本申請案闡述經組態以實施此等編碼及解碼方法之編碼器及解碼器。

在再一態樣中，本申請案闡述儲存電腦可執行程式指令之非暫時性電腦可讀媒體，該等電腦可執行程式指令在執行時組態一處理器以執行所闡述之編碼及/或解碼方法。

依據連同附圖一起對實例之以下說明之一審閱，熟習此項技術者將理解本申請案之其他態樣及特徵。

【實施方式】

現在將以實例方式參考展示本申請案之實例性實施例之隨附圖式。

在不同圖中可能已使用類似元件符號來表示類似組件。

在以下說明中，參考用於視訊編碼之H.264標準及/或在開發中之HEVC標準闡述某些實例性實施例。熟習此項技術者將理解，本申請案並不限於H.264/AVC或HEVC，而是可適用於其他視訊編碼/解碼標準，包含可能的未來標準、多視圖編碼標準、可擴縮視訊編碼標準及可重新組態視訊編碼標準。

在以下說明中，當提及視訊或影像時，可在某種程度上可互換地使用術語圖框、圖像、圖塊(slice)、影像塊(tile)及矩形圖塊群組。熟習此項技術者將瞭解，在H.264標準之情形中，一圖框可含有一或多個圖塊。亦將瞭解，某些編碼/解碼操作在一逐圖框基礎上執行，某些在一逐圖塊基礎上執行，某些在逐圖像基礎上執行，某些在逐影像塊

基礎上執行且某些在逐矩形圖塊群組基礎上執行，此取決於適用影像或視訊編碼標準之特定要求或術語。視情形而定，在任何特定實施例中，適用影像或視訊編碼標準可判定連同以下各項中之哪一者來執行下文所闡述之操作：圖框及/或圖塊及/或圖像及/或影像塊及/或矩形圖塊群組。因此，根據本發明，熟習此項技術者將理解，本文中所闡述之特定操作或程序及對圖框、圖塊、圖像、影像塊、矩形圖塊群組之特定參考是否適用於一給定實施例之圖框、圖塊、圖像、影像塊、矩形圖塊群組或者彼等中之某些或所有。如根據下文之說明將顯而易見，此亦適用於變換單元、編碼單元、編碼單元群組等。

本申請案係關於有效性映射。一有效性映射係映射至或對應於一變換單元之一旗標區塊、矩陣或群組。每一旗標指示變換單元中之對應位置是否含有一非零係數。在現有標準中，此等旗標可稱作有效係數旗標。在現有標準中，每係數存在一個旗標且旗標為一位元，若對應係數係零，則旗標為係零，且若對應係數係非零，則將旗標設定為1。如本文中所使用之術語「有效性映射」意欲指一變換單元之一有效係數旗標矩陣或有序集合，如自下文之說明將理解。

現在參考圖1，其以方塊圖形式展示用於編碼視訊之一編碼器10。亦參考圖2，其展示用於解碼視訊之一解碼器50之一方塊圖。應瞭解，本文中所闡述之編碼器10及解碼器50可各自實施於含有一或多個處理元件及記憶體之一特

殊應用或一般用途計算裝置上。視情形而定，可(舉例而言)憑藉特殊應用積體電路或憑藉可由一般用途處理器執行之所儲存程式指令來實施由編碼器10或解碼器50執行之操作。該裝置可包含額外軟體，包含(舉例而言)用於控制基本裝置功能之一作業系統。慮及以下說明之熟習此項技術者將瞭解在其內可實施編碼器10或解碼器50之裝置及平台之範圍。

編碼器10接收一視訊源12且產生一經編碼位元串流14。解碼器50接收經編碼位元串流14且輸出一經解碼視訊圖框16。編碼器10及解碼器50可經組態以符合若干個視訊壓縮標準而操作。舉例而言，編碼器10及解碼器50可符合H.264/AVC。在其他實施例中，編碼器10及解碼器50可符合其他視訊壓縮標準，包含H.264/AVC標準之演進(如HEVC)。

編碼器10包含一空間預測器21、一編碼模式選擇器20、變換處理器22、量化器24及熵編碼器26。如熟習此項技術者將瞭解，編碼模式選擇器20判定視訊源之適當編碼模式，舉例而言，判定標的圖框/圖塊是I型、P型還是B型及圖框/圖塊內之特定編碼單元(例如，巨集區塊、編碼單元等)是圖框間編碼還是圖框內編碼的。變換處理器22對空間域資料執行一變換。特定而言，變換處理器22應用一基於區塊之變換以將空間域資料轉換成光譜分量。舉例而言，在諸多實施例中使用一離散餘弦變換(DCT)。在某些例項中可使用諸如一離散正弦變換或其他之其他變換。取

決於巨集區塊或編碼單元之大小，在一編碼單元、巨集區塊或子區塊基礎上執行基於區塊之變換。舉例而言，在H.264標準中，一典型 16×16 巨集區塊含有十六個 4×4 變換區塊且對 4×4 區塊執行DCT程序。在某些情形中，變換區塊可係 8×8 ，此意指每巨集區塊存在四個變換區塊。在再一些情形中，變換區塊可係其他大小。在某些情形中，一 16×16 巨集區塊可包含 4×4 變換區塊與 8×8 變換區塊之一非重疊組合。

將基於區塊之變換應用至一像素資料區塊產生一變換域係數集合。在此內容脈絡中，一「集合」係其中係數具有係數位置之一有序集合。在某些例項中，可將變換域係數集合視為一係數「區塊」或矩陣。在本文中之說明中，片語一「變換域係數集合」或一「變換域係數區塊」可互換地使用且意欲指示一有序變換域係數集合。

由量化器24量化變換域係數集合。然後由熵編碼器26編碼經量化係數及相關聯資訊。

經量化變換域係數區塊或矩陣在本文中可稱作一「變換單元」。

經圖框內編碼圖框/圖塊(亦即，類型I)係在不參考其他圖框/圖塊之情況下編碼。換言之，其不採用時間預測。然而，經圖框內編碼圖框確實依賴於圖框/圖塊內之空間預測，如圖1中藉由空間預測器21所圖解說明。亦即，當編碼一特定區塊時，可比較該區塊中之資料與已針對彼圖框/圖塊編碼之區塊內之附近像素之資料。使用一預測演

算法，可將區塊之源資料轉換成殘餘資料。變換處理器22然後編碼殘餘資料。舉例而言，H.264針對4×4變換區塊規定九個空間預測模式。在某些實施例中，可使用九個模式中之每一者來獨立地處理一區塊，且然後使用位元率失真最佳化(rate-distortion optimization)來選擇最佳模式。

H.264標準亦規定使用運動預測/補償來利用時間預測。因此，編碼器10具有一回饋環路，該回饋環路包含一解量化器28、逆變換處理器30及解區塊處理器32。解區塊處理器32可包含一解區塊處理器及一濾波處理器。此等元件反映由解碼器50實施之解碼程序以重現圖框/圖塊。一圖框儲存器34用以儲存經重現圖框。以此方式，運動預測基於將成為解碼器50處之經重新建構圖框之圖框而非基於原始圖框，原始圖框可由於編碼/解碼中所涉及之有損壓縮而不同於經重新建構圖框。出於識別類似區塊之目的，一運動預測器36使用儲存於圖框儲存器34中之圖框/圖塊作為源圖框/圖塊以供與一當前圖框比較。因此，針對對其應用運動預測之巨集區塊或編碼單元，變換處理器22編碼之「源資料」係由運動預測程序產生之殘餘資料。舉例而言，源資料可包含關於參考圖框、一空間位移或「運動向量」及表示參考區塊與當前區塊之間的差異(若存在)之殘餘像素資料之資訊。關於參考圖框及/或運動向量之資訊可不由變換處理器22及/或量化器24處理，而是替代地可與經量化係數一起作為位元串流之部分供應至熵編碼器26以供進行編碼。

熟習此項技術者將瞭解用於實施視訊編碼器之細節及可能的變化。

解碼器 50 包含一熵解碼器 52、解量化器 54、逆變換處理器 56、空間補償器 57 及解區塊處理器 60。解區塊處理器 60 可包含解區塊處理器及濾波處理器。一圖框緩衝器 58 供應經重新建構圖框以供由一運動補償器 62 用於應用運動補償。空間補償器 57 表示自一先前經解碼區塊復原一特定經圖框內編碼區塊之視訊資料之操作。

由熵解碼器 52 接收且解碼位元串流 14 以復原經量化係數。在熵解碼程序期間亦可復原側資訊，若適用，則可將某些側資訊供應至運動補償環路以供在運動補償中使用。舉例而言，熵解碼器 52 可復原經圖框間編碼巨集區塊之運動向量及/或參考圖框資訊。

然後由解量化器 54 解量化經量化係數以產生變換域係數，該等變換域係數然後經受藉由逆變換處理器 56 之一逆變換以重建「視訊資料」。應瞭解，在某些情形中(諸如在一經圖框內編碼巨集區塊或編碼單元之情況下)，經重新形成「視訊資料」係供在相對於圖框內之一先前經解碼區塊之空間補償中使用之殘餘資料。空間補償器 57 自殘餘資料產生視訊資料且自一先前經解碼區塊產生像素資料。在其他情形中(諸如經圖框間編碼巨集區塊或編碼單元)，來自逆變換處理器 56 之經重新形成「視訊資料」係供在相對於來自一不同圖框之一參考區塊之運動補償中使用之殘餘資料。空間補償及運動補償兩者在本文中皆可稱作「預測

操作」。

運動補償器 62 將一參考區塊定位於指定用於一特定經圖框間編碼巨集區塊或編碼單元之圖框緩衝器 58 內。運動補償器 62 基於指定用於經圖框間編碼巨集區塊或編碼單元之參考圖框資訊及運動向量而如此進行。運動補償器 62 然後供應參考區塊像素資料以供與殘餘資料組合以達成用於彼編碼單元/巨集區塊之經重新建構視訊資料。

然後可將一解區塊/濾波程序應用至一經重新建構圖框/圖塊，如藉由解區塊處理器 60 所指示。在解區塊/濾波之後，將圖框/圖塊作為經解碼視訊圖框 16 輸出，(舉例而言) 以供在一顯示裝置上顯示。應理解，諸如一電腦、機上盒、DVD 或藍光播放器及/或行動手持式裝置之視訊播放機器可在於一輸出裝置上顯示之前在一記憶體中緩衝經解碼圖框。

預期，符合 HEVC 之編碼器及解碼器將具有此等相同或類似特徵中之諸多特徵。

有效性映射編碼

如上文所提及，一經量化變換域係數區塊或集合之熵編碼包含針對彼經量化變換域係數區塊或集合編碼有效性映射(例如，一有效係數旗標集合)。有效性映射係指示非零係數顯現於哪些位置(除最後位置之外)中之區塊之一個二進制映射。區塊可具有其與之相關聯之某些特性。舉例而言，區塊可來自一經圖框內編碼圖塊或一經圖框間編碼圖塊。區塊可係一明度區塊或一色度區塊。圖塊之 QP 值可在

圖塊間不同。所有此等因素可對熵編碼有效性映射之最佳方式具有一影響。

根據掃描次序(其可係垂直、水平、對角線、之字形或由適用編碼標準規定之任何其他掃描次序)將有效性映射轉換成一向量。通常以「反向」次序進行掃描，亦即，以最後有效係數開始且以反向掃描次序往回遍及有效性映射工作直至到達[0, 0]處之旗標。然後使用適用內容脈絡自適應編碼方案熵編碼每一有效係數旗標。舉例而言，在諸多應用中可使用一內容脈絡自適應二進制算術編碼(CABAC)方案。其他實施方案可使用藉助二進制化之其他內容脈絡自適應編解碼器。實例包含二進制算術編碼(BAC)、可變至可變(V2V)長度編碼及可變至固定(V2F)長度編碼。在 4×4 及 8×8 映射之情況下，針對每一位元位置指派一內容脈絡。當編碼彼位元位置中之位元(有效係數旗標)時，經指派內容脈絡及至彼點之內容脈絡之歷史判定一最小可能符號(LPS)(或在某些實施方案中一最大可能符號(MPS))之估計之機率。

在現有視訊編碼器中，針對編碼器及解碼器兩者預定內容脈絡指派。舉例而言，在一 4×4 明度區塊之情況下，當前HEVC標準草案規定 4×4 有效性映射中之每一位元位置具有一唯一內容脈絡。排除最後位置，彼意指追蹤15個內容脈絡以用於 4×4 明度有效性映射之編碼。針對每一位元位置，指派至彼位置之內容脈絡判定與彼位置中之一LPS相關聯之估計之機率。然後使用彼估計之機率編碼實際位元

值。最後，基於實際位元值更新指派至彼位置之內容脈絡。在解碼器處，使用相同內容脈絡模型解碼經編碼資料。追蹤每一位元位置之一內容脈絡且使用該內容脈絡來判定解碼資料以復原彼位置之位元之估計之機率。

在 16×16 及 32×32 有效性映射之情況下，一有效性係數之內容脈絡(主要)基於相鄰有效係數旗標值。在用於 16×16 及 32×32 有效性映射之13個內容脈絡當中，存在專用於 $[0, 0]$ 處之位元位置及相鄰位元位置之某些內容脈絡，但大多數有效係數旗標採用取決於相鄰有效係數旗標之累積值之五個內容脈絡中之一者。在此等例項中，一有效係數旗標之正確內容脈絡之判定取決於相鄰位置(通常五個位置，但在某些例項中可能更多或更少)處之有效係數旗標之值之判定及求總和。此涉及多重記憶體存取，此在記憶體頻寬要求上可係高成本的。此外，在諸多例項中， 16×16 及 32×32 有效性映射含有大量零。因此，在編碼及傳輸具有少數係數值之大映射中涉及到一實質成本。

根據本申請案之一項態樣，編碼器及解碼器針對某些變換單元使用多層級有效性映射。在下文所闡述之實例中，多層級有效性映射用於 16×16 及 32×32 大小之變換單元；然而，應理解，在某些實施例中其可用於 8×8 或 64×64 或其他大小之變換單元。

將有效係數旗標分組。每一有效係數旗標歸屬於該等群組中之一者中。為了簡化，在諸多實施例中藉由(在概念上)將變換單元結構劃分或分割成若干區塊來形成該等群

組。舉例而言，可將一 16×16 映射劃分成各自含有十六個係數位置之 4×4 區塊。可將一 32×32 映射劃分成各自含有六十四個係數位置之 8×8 區塊。因此，在有效係數旗標歸屬於呈矩陣結構之此等經定義區塊中之基礎上將該等有效係數旗標分組。

圖3展示一實例性 16×16 變換單元100(經量化變換域係數矩陣)。出於索引目的，可由 $[xC, yC]$ 指定變換單元內之位元位置，其中 $xC=0, 1, 2, \dots, 15$ 且 $yC=0, 1, 2, \dots, 15$ 。使用(舉例而言)一對角線掃描次序，應注意，此實例中之最後有效係數係在 $[12, 10]$ 處，如由元件符號112指示。

現在亦參考圖4，其展示一實例性有效性映射102。有效性映射102含有自圖4中所展示之實例性變換單元產生之有效係數旗標。應注意，一有效係數旗標以自 $[0, 0]$ 直至(但不包含) $[12, 10]$ 處之最後有效係數之掃描次序顯現於每一位元位置處。將在變換單元100中存在一非零係數所針對的每一位元位置處之有效係數旗標設定為1，而將在其處存在一零係數之每一位元位置處之每一有效係數旗標設定為零。

在一項實施例中，可基於將變換單元結構劃分成連續區塊之一均勻劃分來分組有效性映射102(亦即，有效係數旗標集合)。變換單元之大小可判定區塊之大小。在一 16×16 變換單元之情形中，在某些實施例中該等區塊可係 4×4 。在圖4中由劃定 4×4 區塊之線圖解說明該等分組。一較大變

換單元(諸如， 32×32 變換單元)可使其有效係數旗標分組成 4×4 區塊、 8×8 區塊或其他大小之連續區塊。

雖然本文中所給出之實例為了簡化而使用定義為連續正方形區塊之群組，但本申請案不限於正方形群組。在某些實施例中，可將群組形成為矩形區塊。在再一些實施例中，可使用其他形狀。舉例而言，在一對角線掃描次序之情況下，使用由變換單元之對角線圖塊形成之群組可係有利的，在該情形中某些群組可在某種程度上係梯形形狀。

然後可產生對應於群組矩陣之一較高層級有效性映射。該較高層級有效性映射係一有序有效係數群組旗標集合。針對含有至少一個有效係數旗標之每一群組存在一個有效係數群組旗標。含有最後有效係數之群組不必包含於該較高層級有效性映射中，此乃因將已知其含有至少一個非零係數(亦即，最後有效係數)。該有效性映射可稱作層級0或L0映射。該較高層級有效性映射(亦即，含有有效係數群組旗標)可稱作層級1或L1映射。

圖5圖解說明對應於圖4中所展示之實例性有效性映射102之L1較高層級有效性映射104。應注意，L1映射104針對含有至少一個有效係數旗標之每一群組含有一有效係數群組旗標。若該群組內之有效係數旗標中之任一者皆係非零，則將有效係數群組旗標設定為1。否則，將其設定為零。

可由[xCG, yCG]指定該等群組之索引，其中在此實例中xCG=0、1、2、3且yCG=0、1、2、3。含有最後有效係數

之群組係在 $[3, 2]$ 處。 $[3, 3]$ 處之群組不含有任何有效係數旗標，因此其不包含於L1映射中。

在某些實施例中，可以一反向掃描次序將有效係數群組旗標轉換成向量形式。該掃描次序可大體相同於指定與變換單元一起使用之掃描次序。在一項實施例中，有效係數群組旗標可使用可不同於用於變換單元之選定掃描次序之一預定義掃描次序。在某些情形中，L1映射可不包含將具有一假定旗標值之某些群組(如 $[0, 0]$ 群組或最後有效係數群組)，如下文將進一步闡述。

應瞭解，不必自L0映射直接推導L1映射，而是可自以掃描次序掃描變換單元中之係數推導L1映射。

亦將瞭解，在某些實施例中可使用進一步較高層級映射。舉例而言，若變換單元係一 64×64 變換單元，則L1映射可基於將該變換單元劃分成256個 4×4 群組。因此，L1映射將係含有L1群組旗標之一 16×16 映射。可藉由將L1旗標分組成一進一步 4×4 區塊(其中之每一者將對應於來自變換單元之一 16×16 係數群組)集合來產生一進一步L2映射。在其他實施例中可採用額外層級之抽象化及/或細微度。

現在參考圖6，其以流程圖形式展示用於編碼有效係數旗標之一實例性程序200。程序200在操作202中以編碼器判定有效係數旗標及有效係數群組旗標開始。在一項實施例中，編碼器以反向掃描次序掃描變換區塊以判定最後有效係數及有效係數旗標集合。可在相同掃描期間判定有效係數群組旗標(但可在實際實施方案中使用值之某一量之

緩衝，此乃因掃描次序將通常涉及穿越多個區塊；在某些情形中，在編碼器判定其已掃描彼群組之最後係數(例如，結束係數(exit coefficient))時做出對有效係數群組旗標之判定)。在某些實施方案中，編碼器可執行對L0有效性映射或變換單元之一第二掃描以判定有效係數群組旗標。

在操作204中，針對每一有效係數群組旗標，編碼器判定將使用之內容脈絡且然後基於該經判定內容脈絡熵編碼彼有效係數群組旗標。可以一規定次序處理該等有效係數群組旗標。在某些實施例中，該規定次序相同於用於變換單元之掃描次序(或反向掃描次序)。可以任何適合方式結構化內容脈絡之數目及其判定。下文稍後闡述一實例性內容脈絡集合及用於判定有效係數群組旗標之內容脈絡之方法。

已編碼了有效係數群組旗標集合，編碼器然後編碼有效係數旗標。在操作206中，編碼器(以反向掃描次序工作)判定每一有效係數旗標之內容脈絡，且若每一有效係數旗標歸屬於將有效係數群組旗標設定為1所針對的一群組中，則編碼彼有效係數旗標。若將對應有效係數群組旗標設定為零，則不編碼彼群組中之有效係數旗標中之任一者，亦即，在熵編碼程序期間跳過該等有效係數旗標。

因此，在程序200之後，編碼器已產生含有經編碼有效係數群組旗標及歸屬於具有至少一個非零有效係數旗標之一群組中之經編碼有效係數旗標之經編碼資料之一位元串

流。該位元串流不包含來自不具有至少一個非零有效係數旗標之任何群組之任何有效係數旗標。

在解碼器處，需自該位元串流之該經編碼資料重新建構有效係數旗標。現在參考圖7，其以流程圖形式展示用於自經編碼資料之一位元串流重新建構有效係數旗標之一實例性程序300。可透過一網路連接接收該位元串流，亦即，自一電腦可讀媒體(諸如，一記憶體(例如，快閃記憶體等)或一儲存碟(例如，DVD、BluRay™、CD-ROM等))流動接收或讀取。在於一解碼器處重新建構一變換單元之程序中應用程序302。針對序列及每一圖塊或圖片兩者，未展示的係標頭資訊之解碼(取決於使用中之視訊編碼標準之語法)。

在操作302中，自位元串流解碼最後有效係數之位置。可以任何適用語法表示此資訊。某些標準規定將使用矩陣表示法(例如，變換單元內之基於x之位置及基於y之位置)指定最後有效係數；某些標準規定將使用在最後有效係數位置處具有一個1之一個0之向量將最後有效係數發訊號，其中按掃描次序將該向量映射至變換單元。在操作302中可使用用於指定最後有效係數之任何適合語法。

在操作304中，自位元串流解碼有效係數群組旗標。可能已使用由標準指定或在標頭資訊中指定之任何適用二進制化方案熵編碼該等有效係數群組旗標。舉例而言，在某些例項中可使用內容脈絡自適應二進制算數編碼。藉由判定每一旗標位置(較高層級有效性映射(例如，L1有效性映

射中之位元位置))之內容脈絡解碼有效係數群組旗標，且然後自位元串流解碼旗標值且基於該旗標值更新該內容脈絡。有效係數群組旗標集合之大小係已知的，此乃因掃描次序係已知的且已在操作302中識別出最後有效係數；因此，判定出L1有效性映射之大小。在非均等分割之群組之情形中，可在語法中提供群組大小及位置之一適合發訊號。

如上文所提及，每一有效係數群組旗標對應於針對變換單元定義之連續群組中之一各別者。有效係數旗標中之一或多者歸屬於具有一有效係數群組旗標之此等群組中之每一者中。因此，每一有效係數群組旗標對應於一各別有效係數旗標群組。

在解碼有效係數群組旗標集合之後，然後以規定掃描次序執行用於重新建構有效性映射(亦即，有效係數旗標集合)之剩餘操作。在一反向掃描次序之情形中，然後處理自最後有效係數(但不包含彼最後有效係數位置，此乃因已知其含有一非零係數)開始。在操作305中，針對每一有效係數旗標，解碼器判定其對應有效係數群組旗標是否為零。若該對應有效係數群組旗標係非零，則自位元串流解碼一有效係數旗標，如由操作306指示。亦即，若相關聯或對應有效係數群組旗標指示群組可含有至少一個非零係數，則解碼器針對當前位置自位元串流解碼一有效係數旗標。

若相關聯或對應有效係數群組旗標係一個零(亦即，其

指示在群組中不存在非零係數)，則解碼器將當前有效係數旗標設定或重新建構為一個零，如由操作308指示。解碼器不自位元串流解碼當前有效係數旗標。

在操作310處，解碼器判定其是否已到達反向掃描次序之末端(亦即，變換單元之左上角(例如，[0, 0])處之係數)。若如此，則程序300結束；若並非如此，則解碼器在操作312中以反向掃描次序移動至下一位置且重複操作306及308以重新建構彼下一位置之有效係數旗標。

應瞭解，掃描次序通常不導致在移動至下一群組上之前重新建構一群組之所有有效係數旗標。而是，掃描次序(取決於掃描次序及群組幾何形狀)通常跨越群組界限掃描以使得解碼器重新建構來自一個群組之幾個旗標、來自一毗鄰群組之幾個旗標等，從而以反向掃描次序往回工作至[0, 0]位置。

在編碼及解碼程序中可存在經計及以節省位元之特殊情形。舉例而言，如上文所提及，含有最後有效係數之群組將總是具有指示一非零係數之一有效係數群組旗標，以使得有效係數群組旗標不需要被編碼及傳輸至解碼器。編碼器總是編碼彼群組之有效係數旗標，且解碼器經組態以總是解碼彼群組之有效係數旗標。

可包含於某些實施例中之另一特殊情形係總是編碼及解碼第一群組。此群組在變換單元中之[0, 0]處含有DC係數。此群組不含有非零係數之機率係極低的。因此，替代傳輸[0, 0]群組之一有效係數群組旗標，編碼器可經組態

以總是編碼彼群組之有效係數旗標且解碼器可經組態以總是解碼彼群組之有效係數旗標。

可在某些實施例中實施之再一特殊情形亦基於機率。已注意到，當在一特定群組右邊之群組及在該特定群組下方之群組兩者皆含有非零係數時，則彼特定群組含有一非零係數之機率極高。因此，在某些實施例中，編碼器及解碼器可假定具有兩者皆含有非零係數之一右邊相鄰群組及下方相鄰群組之任何群組，則彼群組具有非零係數。因此，關於一特定群組，若將右邊群組之有效係數群組旗標設定為1，且若將下方群組之有效係數群組旗標設定為1，則編碼器不編碼該特定群組之一有效係數群組旗標且總是編碼該特定群組之有效係數旗標。解碼器辨識右邊及下方鄰近者具有指示非零係數之有效係數群組旗標，因此其將自動地假定該特定群組具有非零係數且其將解碼有效係數旗標。

現在參考圖8，其用額外細節展示接續圖7之操作304以反映處置上文所闡述之特殊情形之一實例性實施例。操作304包含其中將含有最後有效係數之群組之有效係數群組旗標設定為1之一操作304-1。在一較早操作(未展示)中自位元串流解碼最後有效係數之位置。

解碼器然後以反向掃描次序移動通過群組。如操作304-2中所提及，解碼器自含有最後有效係數之群組以反向掃描次序移動至下一群組。針對此群組，解碼器評價在當前群組右邊之群組之有效係數群組旗標及在該當前群組下方

之群組之有效係數群組旗標是否等於1。最初，解碼器將由於其剛開始而在右邊及下方不具有旗標，但稍後以掃描次序(無論水平、垂直還是對角線)，解碼器可有時在相對於當前群組之此等位置處具有經重新建構有效係數群組旗標(對於定位於變換單元之底部邊緣處之群組，解碼器可能永遠不會具有下方之一群組之一旗標)。若將彼兩個毗鄰群組設定為1，則亦將當前群組設定為1之機率係充分高的，以至於編碼器及解碼器兩者假定當前群組被設定為1。因此，在操作304-6中，若滿足該特殊情形條件，則解碼器將有效係數群組旗標設定為1。否則，解碼器移動至操作304-4。在另一實施例中，可將此特殊情形修改為基於其他毗鄰群組或其他群組全部之有效係數群組旗標。

在操作304-4中，解碼器自位元串流解碼當前群組之有效係數群組旗標。解碼包含判定內容脈絡且然後根據經判定內容脈絡進行解碼。解碼可基於二進制算數編碼(BAC)或其他二進制化編碼/解碼程序。

在操作304-5中，解碼器以反向掃描次序判定此是否為倒數第二個群組。若不是，則解碼尚未結束，因此解碼器往回循環至操作340-2以按反向掃描次序前進至下一群組。若以反向掃描次序該下一群組係倒數第二個群組，則解碼器繼續移動至操作304-7，在操作304-7處解碼器將最後群組(亦即，群組[0, 0])之有效係數群組旗標設定為1。此基於以下特殊情形：其中彼特定群組總是由編碼器及解碼器假定為具有至少一個非零係數，因此總是將該有效係

數群組旗標預設為1，以使得總是編碼及解碼彼群組之有效係數旗標。在此操作之後，解碼器繼續進行至操作306或308(圖7)。

應瞭解，圖7及圖8中所圖解說明之以上實例性程序將L1有效性映射(有效係數群組旗標)之解碼及L0有效性映射(有效係數旗標)之解碼展示為其中完全解碼L1有效性映射且然後解碼L0有效性映射之兩階段程序。在某些實施例中，情形可係如此；然而，在某些其他實施例中，該等解碼程序可部分地交織。亦即，L0映射之解碼可在完全解碼L1映射之前開始。應瞭解，在某些實施例中，L0有效性映射之解碼可在一重新建構第一有效係數群組旗標就開始。

在某一實施例中，可取決於(舉例而言)圖像類型而開始或停止多層級有效性映射編碼。舉例而言，可針對I圖像及P圖像啟用多層級有效性映射編碼，但針對B圖像停用多層級有效性映射編碼。

內容脈絡模型化

為了改良編碼效率，BAC引擎(或其他熵編碼/解碼引擎)使用內容脈絡。本申請案提出使用四個新內容脈絡用於編碼有效係數群組旗標。兩個內容脈絡用於明度編碼/解碼，且兩個內容脈絡用於色度編碼/解碼。

假定使用反向掃描次序(在其他實施方案中可使用其他規定次序)。判定該兩個內容脈絡中之哪一者適用於一給定有效係數群組旗標可如下發生。若在當前群組右邊之毗鄰群組之有效係數群組旗標係零，且在當前群組下方之毗

鄰群組之有效係數群組旗標係零，則用於編碼當前群組之有效係數群組旗標之內容脈絡係0。否則，該內容脈絡係1。若彼兩個毗鄰群組之旗標不可用，則出於內容脈絡判定之目的而假定不可用旗標=0。

注意，若使用正向掃描次序，則可改變內容脈絡模型以使用在當前群組左邊之毗鄰群組之有效係數群組旗標及在當前群組上方之毗鄰群組之有效係數群組旗標來判定該內容脈絡。

內容脈絡判定程序亦可包含特殊情形。舉例而言，可總是將內容脈絡1指派給左上部群組。

存在可使用之用於判定內容脈絡之其他可能內容脈絡模型及方法。下文給出某些實例。

為定義表示法，令 $L[i]$ 表示層級 L 處之係數群組 i 之有效旗標且令 N 表示層級 L 處之係數群組之數目。大體而言，對於一給定 L 及係數群組 i ，使用 i 與所有可用 $L[j]$ 之一函數 $c(*)$ 來判定 $L[i]$ 之一內容脈絡 C_i 。因此，藉由以下給出該內容脈絡：

$$C_i = c(i, L[0], L[1], \dots, L[N - 1])$$

其中 $j \neq i$ 。注意，為了使用 $L[j]$ 來判定 $L[i]$ 之一內容脈絡， $L[j]$ 自身必須為可用的。因此，選定掃描次序必須保證先前已判定用於 $c(*)$ 中之任何 $L[j]$ 。

在類似於上文所闡述之內容脈絡判定模式之一實施例中，可藉由以下判定該內容脈絡：

$$C_i = c(i, L0[0], L0[1], \dots, L0[15]) \\ = \text{sum}\{b_j * L0[j]\}$$

其中 $j=0, 1, \dots, N$ 且 $j \neq i$ ，若係數群組 j 係係數群組 i 之右邊或下方鄰近者則 $b_j=1$ ，且否則 $b_j=0$ 。此特定實施例具有 3 個內容脈絡 (若不同 3 個內容脈絡用於色度，則此特定實施例具有 6 個內容脈絡)。

藉由以下給出 $c(*)$ 之另一實施例：

$$C_i = c(i, L[0], L[1], \dots, L[N - 1]) \\ = \text{sum}\{b_j * L[j]\}$$

其中 $j=0, 1, \dots, N$ 且 $j \neq i$ ，若係數群組 j 係已判定之 i 之任何相鄰係數群組，則 b_j 係非零，否則 $b_j=0$ 。在此實施例中，加權係數 b_j 可不必係常數。

$c(*)$ 之另一實施例忽略 L 處之其他係數群組之有效係數群組旗標且僅基於當前係數群組之位置 i 而判定內容脈絡。此可表達為：

$$C_i = c(i, L[0], L[1], \dots, L[N - 1]) \\ = i$$

可與多層級有效性映射一起使用用於判定內容脈絡之其他內容脈絡模型及程序。

位元率失真最佳化量化

某些編碼程序採用位元率失真最佳化量化 (RDOQ) 或有時稱作「軟量化」之程序。RDOQ 係一種基於一位元率失真最佳化表達來判定最佳經量化變換域係數之程序。因此，由 RDOQ 產生之經量化變換域係數可或可不相同於透

過正常變換及量化程序達成之經量化變換域係數。在某些情形中，可已藉由RDOQ程序修改係數值，此乃因所得失真已經判定比傳輸成本中之結果節省成本較少。

RDOQ程序通常評估判定位元率分量之四個成本。四個位元率成本包含最後位置位元率(last position rate)、有效位元率(significance rate)(L0位元率)、係數位元率(coefficient rate)及經編碼區塊參數(CBP)位元率。為了實施多層級有效性映射，修改RDOQ以在RDOQ計算中亦包含較高層級有效位元率(例如，L1位元率)可係有利的。

在一項實施例中，可修改RDOQ程序以關於有效性映射執行一個兩階段RDOQ。首先，應用RDOQ程序以判定最佳最後位置及係數值及因此判定L1有效係數旗標。在第一階段中，在固定最後位置之情況下，可然後關於L1位元率再次應用RDOQ程序以判定是否存在用於將任何係數歸零之一位元率失真(RD)成本理由。

圖9以流程圖形式展示用於編碼多層級有效性映射之一實例性RDOQ程序400。程序400使用RDOQ來獲得最佳經量化變換係數且判定最後有效係數(亦即，一L0 RDOQ)之位置。程序400然後固定最後位置且調整當前RD成本以計及有效性映射之額外層級之效應。然後，其使用一貪婪方法來進一步最佳化變換係數。

操作402反映使用RDOQ來獲得最佳經量化變換域係數，此提供一最後有效係數位置。操作402基於傳輸最後位置、對應於最佳係數之有效性映射、係數值及CBP之位元

率而產生某一RD成本。

在操作404中，固定最後有效係數位置。亦即，最後群組將含有一非零係數，亦即，將最後有效係數群組旗標固定為1。編碼器然後貪婪地判定是否藉由將其他群組中之係數歸零而得到成本節省。在某些實施例中，可以反向掃描次序執行程序400，但可以另一次序處理程序400。

在操作406中，以作為當前群組之倒數第二個群組開始，編碼器判定當前群組是否具有一有效係數群組旗標=1。若並非如此，則該群組已僅含有零且編碼器跳過至下一群組。若有效係數群組旗標=1，則編碼器計算一RD成本，若當前群組中之所有係數皆係零，則會產生該成本。在操作408中，編碼器評價新近計算之RD成本是否比當前RD成本更佳(例如，較少)。若如此，則在操作410中，將當前群組中之所有係數歸零，且更新當前RD成本以反映該改變。在操作412中，編碼器評價是否關於L1 RDOQ進行操作(例如，該操作是否已到達恰在 $[0, 0]$ 群組之前的群組)(若編碼器及解碼器經組態以假定在彼群組中存在至少一個非零係數(如上文所概述之特殊情形中所闡述)，則不將 $[0, 0]$ 群組歸零)。若存在將評價之其他群組，則程序400在操作414處繼續，在操作414處編碼器移動至下一群組(在某些實施例中，使用反向掃描次序)。

現在將憑藉一實例圖解說明RDOQ程序。將連同圖3、圖4及圖5一起再次參考上文所給出之實例。在L1 RDOQ之前但在L0 RDOQ之後，在圖3中展示最佳經量化變換域係

數。在圖4中展示對應L0有效性映射，且在圖5中展示L1有效性映射。

L1 RDOQ程序可產生(舉例而言)圖10中所展示之一最佳L0有效性映射500及圖11中所展示之相關聯或對應L1有效性映射502。

應注意，最後有效群組(亦即，[3, 2]群組)中之有效係數旗標未改變。然而，已將[3, 0]群組、[0, 3]群組及[1, 3]群組全部歸零。因此，亦已將此三個群組之對應有效係數群組旗標改變為零，如圖11中所圖解說明。結果係編碼器將不必編碼此三個群組。因將在彼等群組中得到之較少係數歸零導致之失真比不上減小經編碼位元之數目之成本節省，如藉由RDOQ評價判定。

在一項可能實施例中，RDOQ程序可經延伸以判定當前TU之最佳係數群組大小。在此實施例中，將程序400重複多個回合，其中每一回合假定一不同係數群組大小且其中操作410經修改以使得實際上未將變換係數設定為0。基本上，在每一回合中，此經修改RDOQ程序計算一特定係數群組大小之RD成本。在所有回合已完成之後，RDOQ選擇產生最小RD成本之係數群組大小且最後視需要將任何變換係數設定為0。編碼器將最佳係數群組大小之值編碼至位元串流中以使得其可由解碼器獲得及使用。

被測試之係數群組大小可基於變換單元大小。舉例而言，一 32×32 變換單元可測試群組大小 8×8 、 4×4 及 2×2 。待測試之群組可係可選擇的，且編碼器可指示(舉例而言，

在序列標頭中)將針對每一變換單元大小測試何種群組大小。舉例而言，假設編碼器及解碼器已同意針對 16×16 TU，經修改RDOQ將測試兩個不同係數群組大小：分別由1及0表示之 2×2 及 4×4 。若經修改RDOQ判定 2×2 係最佳的，則編碼器在significant_coeffgroup_flag's之前將一值格1編碼至位元串流中。解碼器在significant_coeffgroup_flag's之前解碼此值格且知曉當前TU之係數群組大小係 2×2 。

實例性語法

下文提供用於實施多層級有效性映射之一實例性語法。此實例性語法僅係一項可能實施方案。

有效係數群組旗標可表示及定義為：

significant_coeffgroup_flag[xCG][yCG]

此旗標針對當前 16×16 或 32×32 變換區塊內之係數群組位置(xCG, yCG)，將位置(xCG, yCG)處之對應係數群組是否具有非零係數規定為如下：

若significant_coeffgroup_flag[xCG][yCG]等於0，則將位置(xCG, yCG)處之係數群組中之非零係數之數目設定為等於0；

否則(significant_coeffgroup_flag[xCG][yCG]等於1)，則除下文所定義之特殊情形之外，位置(xCG, yCG)處之係數群組中之非零係數之數目係非零。

特殊情形係定義為如下：

1. 將以掃描次序在第一係數群組位置(0, 0)處之significant_coeffgroup_flag[0][0]推斷為等於1。

2. 若 $\text{significant_coeffgroup_flag}[xCG][yCG+1]=1$ 且 $\text{significant_coeffgroup_flag}[xCG+1][yCG]=1$ ，則將以掃描次序在係數群組位置 (xCG, yCG) 處之 $\text{significant_coeffgroup_flag}[xCG][yCG]$ 推斷為等於 1。

當 $\text{significant_coeffgroup_flag}[xCG][yCG]$ 不存在時，將其推斷為等於 0。

在某些實施例中， $\text{significant_coeffgroup_flag}[xCG][yCG]$ 不適用於 4×4 及 8×8 變換區塊。

以下虛擬程式碼圖解說明用於重新建構經量化變換域係數(殘餘)之解碼程序內之多層級有效性映射之一項實例性實施方案。

應注意，虛擬程式碼之第一部分包含解碼最後有效係數位置。若變換單元係 16×16 或更大(如若 $(\log_2 \text{TrafoSize} > 3)$ 所指示)，則判定群組之數目。

residual_coding_cabac(x0, y0, log2TrafoSize, trafoDepth, scanIdx, cIdx) {	描述符
last_significant_coeff_x	ae(v)
last_significant_coeff_y	ae(v)
numCoeff = 0	
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][numCoeff][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][numCoeff][1]	
if (log2TrafoSize > 3) {	
log2CoeffGroupSize = log2trafoSize-2	
numCoeffGroup = 0	
for (xCG = 0; xCG < 4; xCG++)	
for (yCG = 0; yCG < 4; yCG++)	
numNonZeroSB [xCG][yCG] = 0	
}	
while((xC != last_significant_coeff_x) (yC != last_significant_coeff_y)) {	
numCoeff++	
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][numCoeff][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][numCoeff][1]	
if (log2TrafoSize > 3) {	
if ((xC % (1 << log2CoeffGroupSize) == 0) && (yC % (1 << log2CoeffGroupSize) == 0))	
numCoeffGroup++	
}	
}	
if (log2TrafoSize > 3) {	
xCGLast = last_significant_coeff_x / (1 << log2CoeffGroupSize)	
yCGLast = last_significant_coeff_y / (1 << log2CoeffGroupSize)	
significant_coeffgroup_flag[xCGLast][yCGLast] = 1	
significant_coeffgroup_flag[0][0] = 1	
for(n = numCoeffGroup; n > 0; n--) {	
xC = ScanOrder[0][0][scanIdx][n][0]	
yC = ScanOrder[0][0][scanIdx][n][1]	
if (!(xC == xCGLast && yC == yCGLast))	
significant_coeffgroup_flag[xCG][yCG]	ae(v)
}	
}	
for(n = numCoeff - 1; n >= 0; n--) {	
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][n][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][scanIdx][n][1]	
if (log2TrafoSize > 3) {	
xC = xC >> (log2CoeffGroupSize)	

yCG = yC >>(log2CoeffGroupSize)	
if(significant_coeff_group_flag[xCG][yCG]) {	
if((xC % (1<< log2CoeffGroupSize) == 0) && (yC % (1<< log2CoeffGroupSize) == 0) && (numNonZeroSB[xCG][yCG] == 0) && n != 0)	
significant_coeff_flag[xC][yC] = 1	
else {	
significant_coeff_flag[xC][yC]	ae(v)
if(significant_coeff_flag[xC][yC])	
numNonZeroSB[xCG][yCG]++	
}	
}	
else {	
significant_coeff_flag[xC][yC] = 0	
}	
} else	
significant_coeff_flag[xC][yC]	ae(v)
}	
numLastSubset = numCoeff >> 4	
for(i = numLastSubset - 1; i >= 0; i--) {	
offset = i << 4	
for(n = 15; n >= 0; n--) {	
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][1]	
if(significant_coeff_flag[xC][yC])	
coeff_abs_level_greater1_flag[n]	ae(v)
}	
for(n = 15; n >= 0; n--) {	
if(coeff_abs_level_greater1_flag[n])	
coeff_abs_level_greater2_flag[n]	ae(v)
}	
for(n = 15; n >= 0; n--) {	
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][1]	
if(significant_coeff_flag[xC][yC]) {	
coeff_sign_flag[n]	ae(v)
}	
}	
for(n = 15; n >= 0; n--) {	
if(coeff_abs_level_greater2_flag[n])	
coeff_abs_level_minus3[n]	ae(v)
xC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][0]	
yC = ScanOrder[log2TrafoSize - 2][log2TrafoSize - 2][0][n + offset][1]	
if(significant_coeff_flag[xC][yC]) {	
transCoeffLevel[x0][y0][trafoDepth][cIdx][n + offset] =	
(coeff_abs_level_minus3[n] + 3) * (1 - 2 * coeff_sign_flag[n])	
} else	
transCoeffLevel[x0][y0][trafoDepth][cIdx][n + offset] = 0	
}	
}	
}	

前述虛擬程式碼展示上文結合圖7一起闡述之實例性程序300之一項實例性實施方案。

現在參考圖 12，其展示一編碼器 900 之一實例性實施例之一簡化方塊圖。編碼器 900 包含一處理器 902、記憶體 904 及一編碼應用程式 906。編碼應用程式 906 可包含儲存於記憶體 904 中且含有用於組態處理器 902 以執行諸如本文中所闡述之彼等操作之操作之指令之一電腦程式或應用程式。舉例而言，編碼應用程式 906 可根據本文中所闡述之多層級有效性映射程序編碼及輸出經編碼之位元串流。應理解，編碼應用程式 906 可儲存於一電腦可讀媒體中，諸如一光碟、快閃記憶體裝置、隨機存取記憶體、硬碟機等。

現在亦參考圖 13，其展示一解碼器 1000 之一實例性實施例之一簡化方塊圖。解碼器 1000 包含一處理器 1002、一記憶體 1004 及一解碼應用程式 1006。解碼應用程式 1006 可包含儲存於記憶體 1004 中且含有用於組態處理器 1002 以執行諸如本文中所闡述之彼等操作之操作之指令之一電腦程式或應用程式。解碼應用程式 1006 可包含經組態以基於如本文中所闡述之多層級有效性映射來重新建構殘餘資料之一熵解碼器。應理解，解碼應用程式 1006 可儲存於一電腦可讀媒體中，諸如一光碟、快閃記憶體裝置、隨機存取記憶體、硬碟機等。

應瞭解，根據本申請案之解碼器及/或編碼器可實施於若干種計算裝置中，包含(但不限於)伺服器、經適地程式化之一般用途電腦、音訊/視訊編碼及播放裝置、電視機上盒、電視廣播設備及行動裝置。可憑藉含有用於組態

一處理器以實施本文中所闡述之功能之指令之軟體來實施解碼器或編碼器。該等軟體指令可儲存於任何適合非暫時性電腦可讀記憶體上，包含CD、RAM、ROM、快閃記憶體等。

應理解，可使用標準電腦程式設計技術及語言來達成本文中所闡述之編碼器及模組、常式、處理程序、執行緒或實施用於組態編碼器之所闡述方法/程序之其他軟體組件。本申請案不限於特定處理器、電腦語言、電腦程式設計慣例、資料結構、其他此等實施方案細節。熟習此項技術者將認識到，所闡述之程序可實施為儲存於揮發性或非揮發性記憶體中之電腦可執行碼之一部分、一特殊應用積體晶片(ASIC)之部分等。

可對所闡述之實施例做出某些調適及修改。因此，將上文所論述之實施例視為說明性而非限制性。

【圖式簡單說明】

- 圖1以方塊圖形式展示用於編碼視訊之一編碼器；
- 圖2以方塊圖形式展示用於解碼視訊之一解碼器；
- 圖3展示含有經量化變換域係數之一實例性變換單元；
- 圖4展示含有圖3之變換單元之有效係數旗標之一實例性有效性映射；
- 圖5展示含有圖4之有效性映射之有效係數群組旗標之一實例性L1有效性映射；
- 圖6以流程圖形式展示編碼有效係數旗標之一實例性方法；

圖 7 以流程圖形式展示用於自經編碼資料重新建構一有效性映射之一實例性方法；

圖 8 以流程圖形式展示用於解碼及重新建構有效係數群組旗標之一實例性程序；

圖 9 以流程圖形式展示用於多層級有效性映射編碼之一實例性位元率失真最佳化量化程序；

圖 10 展示在應用 L1 RDOQ 之後的圖 4 之實例性 L0 有效性映射；

圖 11 展示對應於在 L1 RDOQ 之後的圖 10 之 L0 有效性映射之 L1 有效性映射；

圖 12 展示一編碼器之一實例性實施例之一簡化方塊圖；
且

圖 13 展示一解碼器之一實例性實施例之一簡化方塊圖。

【主要元件符號說明】

- | | |
|----|--------------|
| 10 | 編碼器 |
| 12 | 視訊源 |
| 14 | 經編碼位元串流/位元串流 |
| 16 | 經解碼視訊圖框 |
| 20 | 編碼模式選擇器 |
| 21 | 空間預測器 |
| 22 | 變換處理器 |
| 24 | 量化器 |
| 26 | 熵編碼器 |
| 28 | 解量化器 |

- 30 逆變換處理器
- 32 解區塊處理器
- 34 圖框儲存器
- 36 運動預測器
- 50 解碼器
- 52 熵解碼器
- 54 解量化器
- 56 逆變換處理器
- 57 空間補償器
- 58 圖框緩衝器
- 60 解區塊處理器
- 62 運動補償器
- 100 變換單元/經量化變換域係數矩陣
- 102 有效性映射/有效係數旗標集合
- 104 L1較高層級有效性映射/L1映射
- 500 最佳L0有效性映射
- 502 L1有效性映射
- 900 編碼器
- 902 處理器
- 904 記憶體
- 906 編碼應用程式
- 1000 解碼器
- 1002 處理器
- 1004 記憶體
- 1006 解碼應用程式

七、申請專利範圍：

1. 一種自經編碼資料之一位元串流重新建構一變換單元之有效係數旗標之方法，該變換單元被分割成若干非重疊有效係數旗標群組，該方法包括：

藉由將至少一個有效係數群組旗標之值設定為1來重新建構有效係數群組旗標，其中每一有效係數群組旗標對應於一各別有效係數旗標群組；及

藉由以下操作來重新建構每一有效係數旗標

若該有效係數旗標在具有係非零之對應有效係數群組旗標之一群組中，則自該位元串流解碼彼有效係數旗標，且

若該有效係數旗標在具有係零之對應有效係數群組旗標之一群組中，則將彼有效係數旗標設定為零。

2. 如請求項1之方法，其中重新建構該等有效係數群組旗標包含：自該位元串流解碼除該至少一個有效係數群組旗標之外的該等有效係數群組旗標中之至少一者。
3. 如請求項1或2之方法，其中將至少一個有效係數群組旗標之該值設定為1包含：將含有一最後有效係數之有效係數旗標之群組之有效係數群組旗標設定為1。
4. 如請求項1至3中任一項之方法，其中將至少一個有效係數群組旗標之該值設定為1包含：將含有一DC係數之有效係數旗標之群組之有效係數群組旗標設定為1。
5. 如請求項1至4中任一項之方法，其中將至少一個有效係數群組旗標之該值設定為1包含：若一有效係數群組旗

標對應於具有其兩者皆具有包含為1之經重新建構值之對應有效係數群組旗標之兩個鄰近群組的一群組，則將彼有效係數群組旗標設定為1。

6. 如請求項5之方法，其中該等群組係連續區塊，且其中該兩個鄰近群組包含一右邊鄰近區塊及一下方鄰近區塊。
7. 如請求項1至6中任一項之方法，其中重新建構有效係數群組旗標進一步包括：針對以一正向掃描次序在含有一最後有效係數之該群組之後的每一有效係數群組旗標，將有效係數群組旗標之值設定為零。
8. 如請求項2之方法，其中自該位元串流解碼該等有效係數群組旗標中之至少一者包括：解碼尚未設定的該變換單元之所有有效係數群組旗標。
9. 如請求項1至8中任一項之方法，其中該變換單元係 16×16 ，且其中每一群組係一 4×4 區塊。
10. 如請求項1至8中任一項之方法，其中該變換單元係 32×32 ，且其中每一群組係一 8×8 區塊。
11. 如請求項1至10中任一項之方法，其中重新建構該等有效係數群組旗標中之一者包含：基於針對兩個鄰近群組重新建構之該等有效係數群組旗標，判定彼有效係數群組旗標之一內容脈絡。
12. 如請求項11之方法，其中該兩個鄰近群組包含一右邊鄰近者及一下方鄰近者，且其中若該兩個鄰近群組之該等有效係數群組旗標兩者皆係0，則該內容脈絡係0，且其

中否則該內容脈絡係1。

13. 如請求項1至12中任一項之方法，其中自該位元串流解碼之該等群組中之一者中之所有該等有效係數旗標之該等值皆係零，但將彼群組之該對應有效係數群組旗標設定為1。
14. 如請求項1至13中任一項之方法，其中該視訊位元串流符合以下各項中之一者：一可擴縮視訊編碼標準、一多視圖編碼標準、一可重新組態視訊編碼標準及一高效視訊編碼標準。
15. 如請求項1至14中任一項之方法，其中每一群組具有不同於其寬度之一高度。
16. 如請求項1至15中任一項之方法，其中每一群組含有十六個有效係數旗標，且其中若該有效係數旗標在具有係非零之對應有效係數群組旗標之一群組中則自該位元串流解碼彼有效係數旗標包括：

若針對彼群組解碼之前十五個有效係數旗標皆係零，則將彼群組中之一第十六個有效係數旗標設定為1。

17. 一種用於解碼經編碼資料之一位元串流以重新建構一變換單元之有效係數旗標之解碼器，該解碼器包括：
 - 一處理器；
 - 一記憶體；及
 - 一解碼應用程式，其儲存於記憶體中且含有用於組態該處理器以執行如請求項1至16中任一項之方法之指令。

18. 一種用於編碼一變換單元之有效係數旗標之方法，該變換單元被分割成若干非重疊有效係數旗標群組，該方法包括：

編碼有效係數群組旗標，其中每一有效係數群組旗標對應於該等有效係數旗標群組中之一各別者，且其中將該有效係數群組旗標設定為零以指示彼對應群組不含有非零有效係數旗標，且其中該編碼不包含經推斷為1之至少一個有效係數群組旗標而無論其對應群組是否含有非零有效係數旗標；及

針對該等有效係數旗標中之每一者，

若該有效係數旗標在其對應有效係數群組旗標係非零所針對的該等群組中之一者中，則編碼彼有效係數旗標，且若該有效係數旗標在該對應有效係數群組旗標係零所針對的該等群組中之一者中，則不編碼彼有效係數旗標。

19. 一種用於編碼一變換單元之有效係數旗標之編碼器，該編碼器包括：

一處理器；

一記憶體，其儲存有效性映射；及

一編碼應用程式，其儲存於記憶體中且含有用於組態該處理器以執行如請求項18之方法之指令。

20. 一種非暫時性處理器可讀媒體，其儲存在執行時組態一或多個處理器以執行如請求項1至16或19中任一項之方法之處理器可執行指令。

八、圖式：

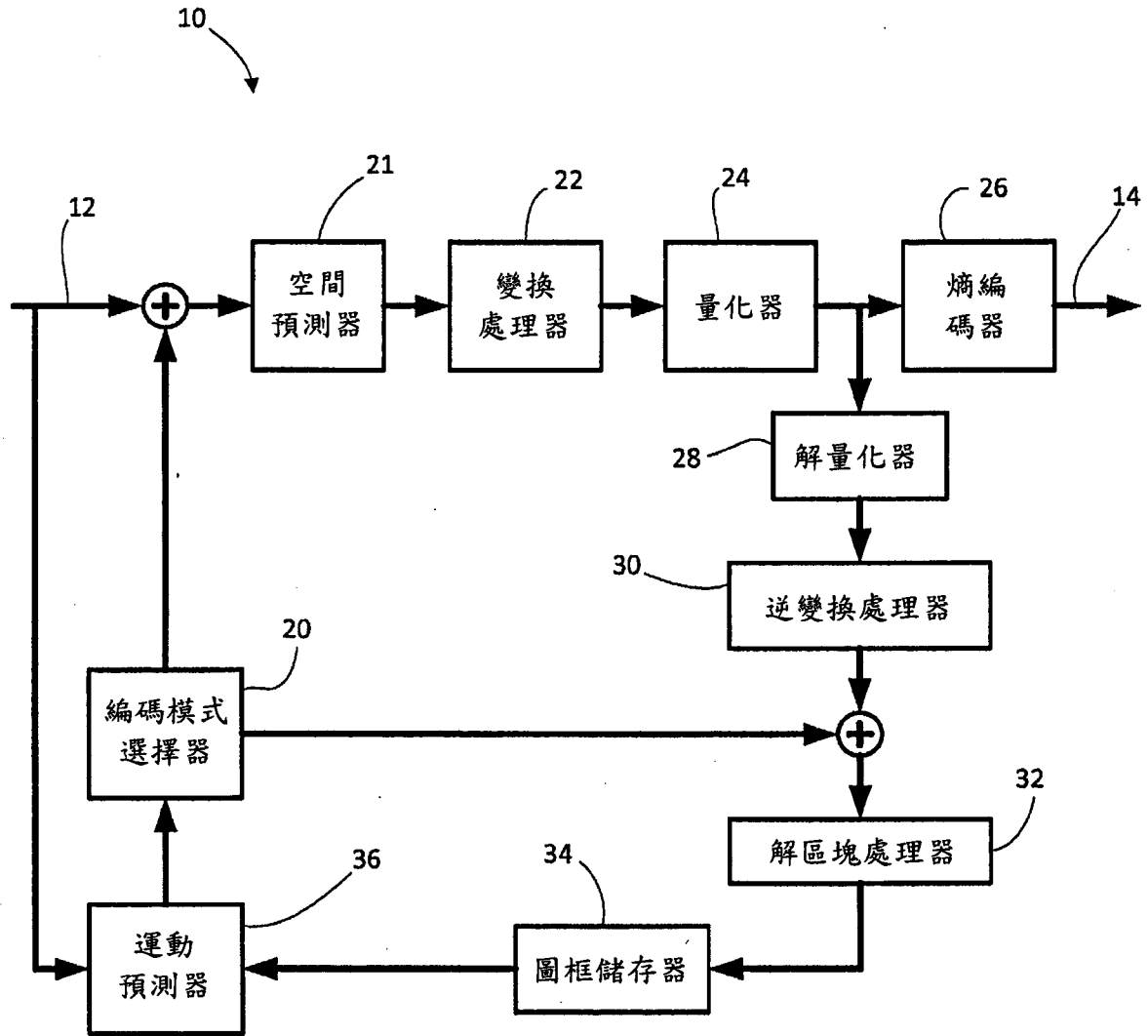


圖 1

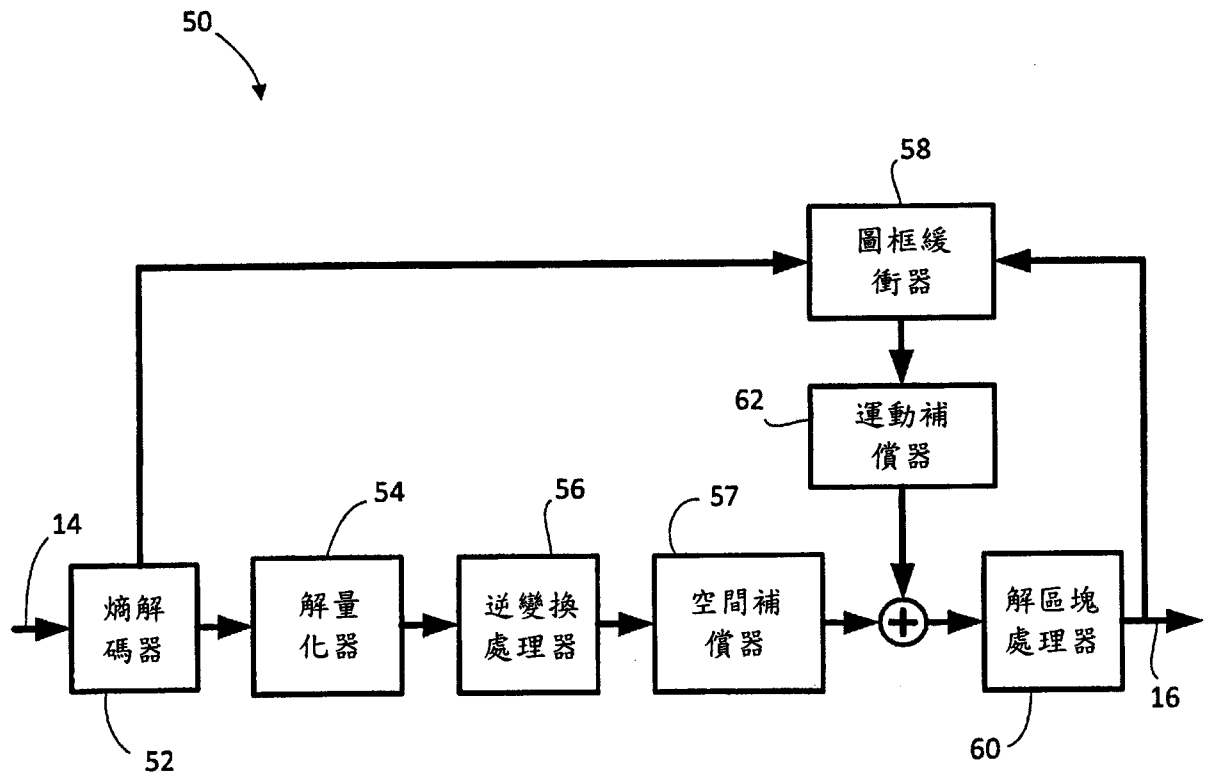


圖 2

xC

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	5	2	0	1	1	0	0	0	1	1	1	0	0	0	0	0
1	5	0	2	1	2	0	1	1	0	2	1	0	1	0	0	0
2	0	1	0	0	1	0	1	1	0	1	1	1	0	0	0	0
3	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0	0
4	1	1	0	0	1	0	1	0	0	1	0	1	0	0	0	0
5	2	1	0	1	0	0	1	0	1	0	0	1	0	0	0	0
6	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
7	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
8	0	1	1	1	0	1	0	1	1	1	1	0	0	1	0	0
9	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0
10	1	0	0	1	0	0	0	0	1	1	0	0	1	0	0	0
11	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
12	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
13	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

yC

100

112

圖 3

xC

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	0	1	1	0	0	0	1	1	1	0	0	0	0	0
1	1	0	1	1	1	0	1	1	0	1	1	0	1	0	0	0
2	0	1	0	0	1	0	1	1	0	1	1	1	0	0	0	0
3	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0	0
4	1	1	0	0	1	0	1	0	0	1	0	1	0	0	0	0
5	1	1	0	1	0	0	1	0	1	0	0	1	0	0	0	0
6	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
7	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
8	0	1	1	1	0	1	0	1	1	1	1	0	0	1	0	0
9	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0
10	1	0	0	1	0	0	0	0	1	1	0	0	1	0	0	0
11	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
12	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
13	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

102

圖 4

104

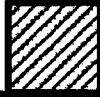
		xCG			
		0	1	2	3
yCG	0	0	1	1	1
	1	1	1	1	0
	2	1	1	1	1
	3	1	1	0	

圖 5

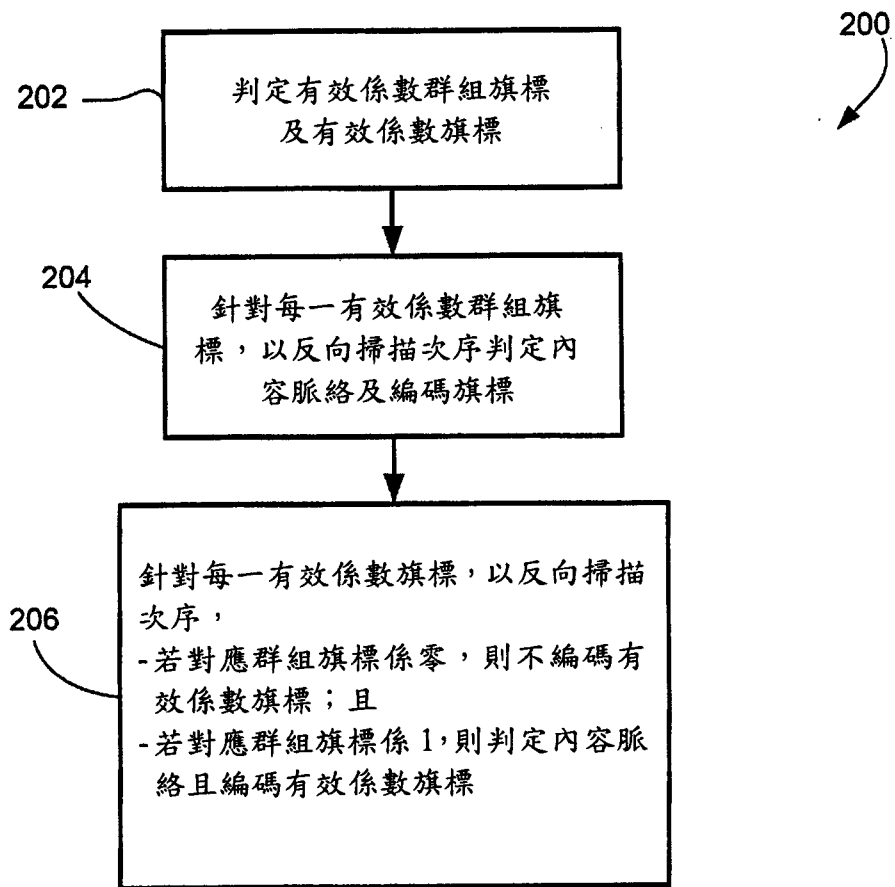


圖 6

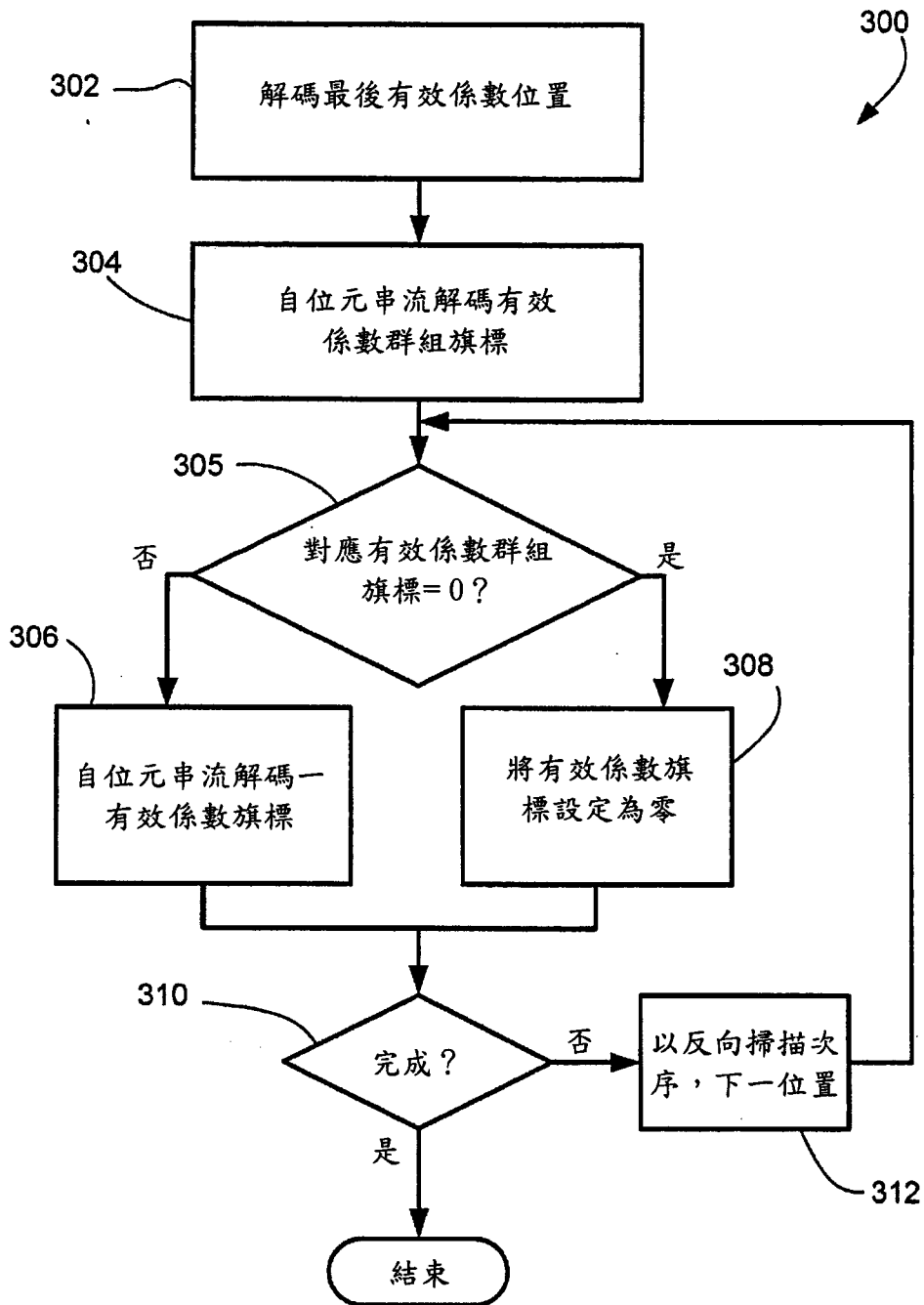


圖 7

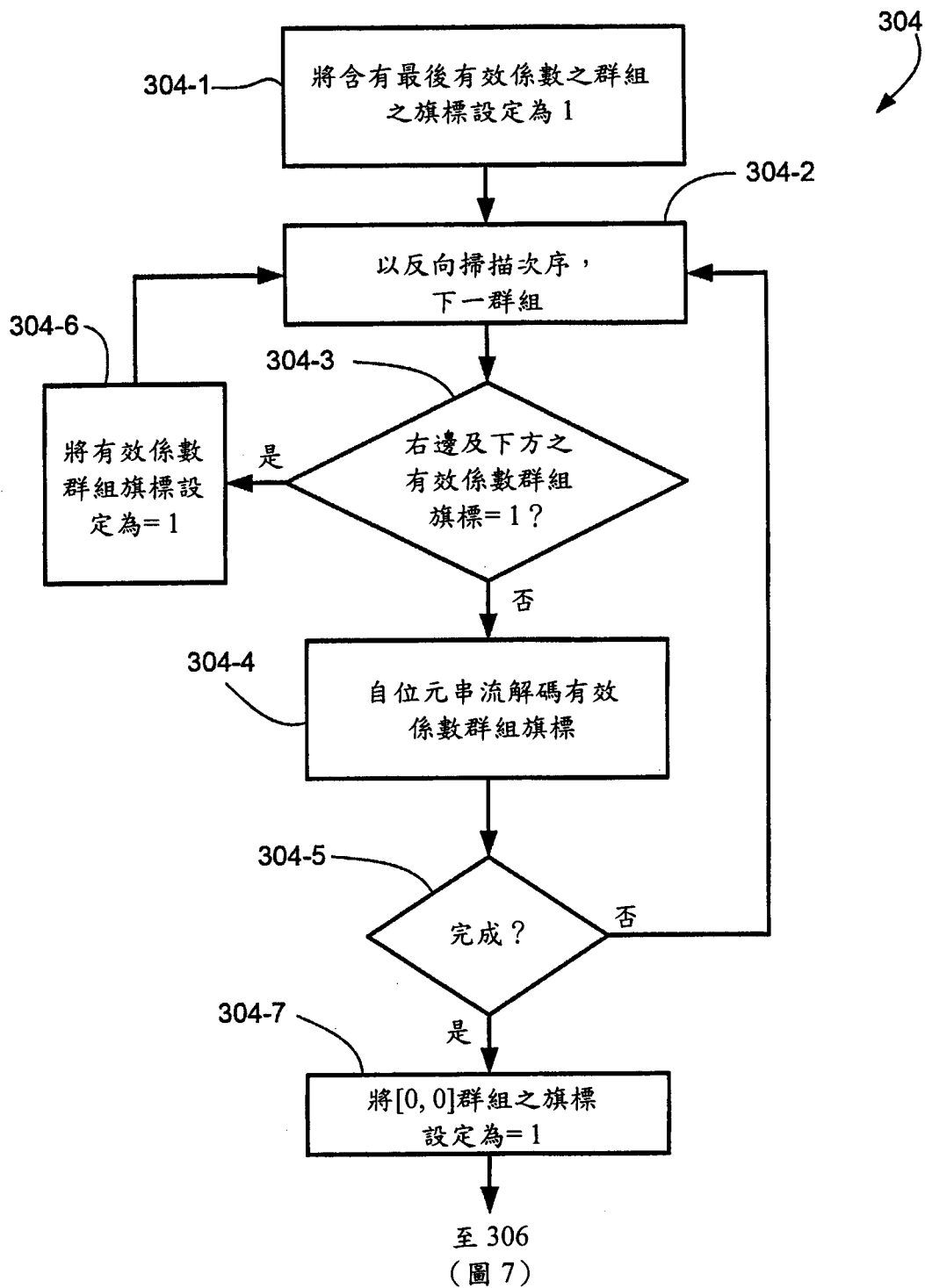


圖 8

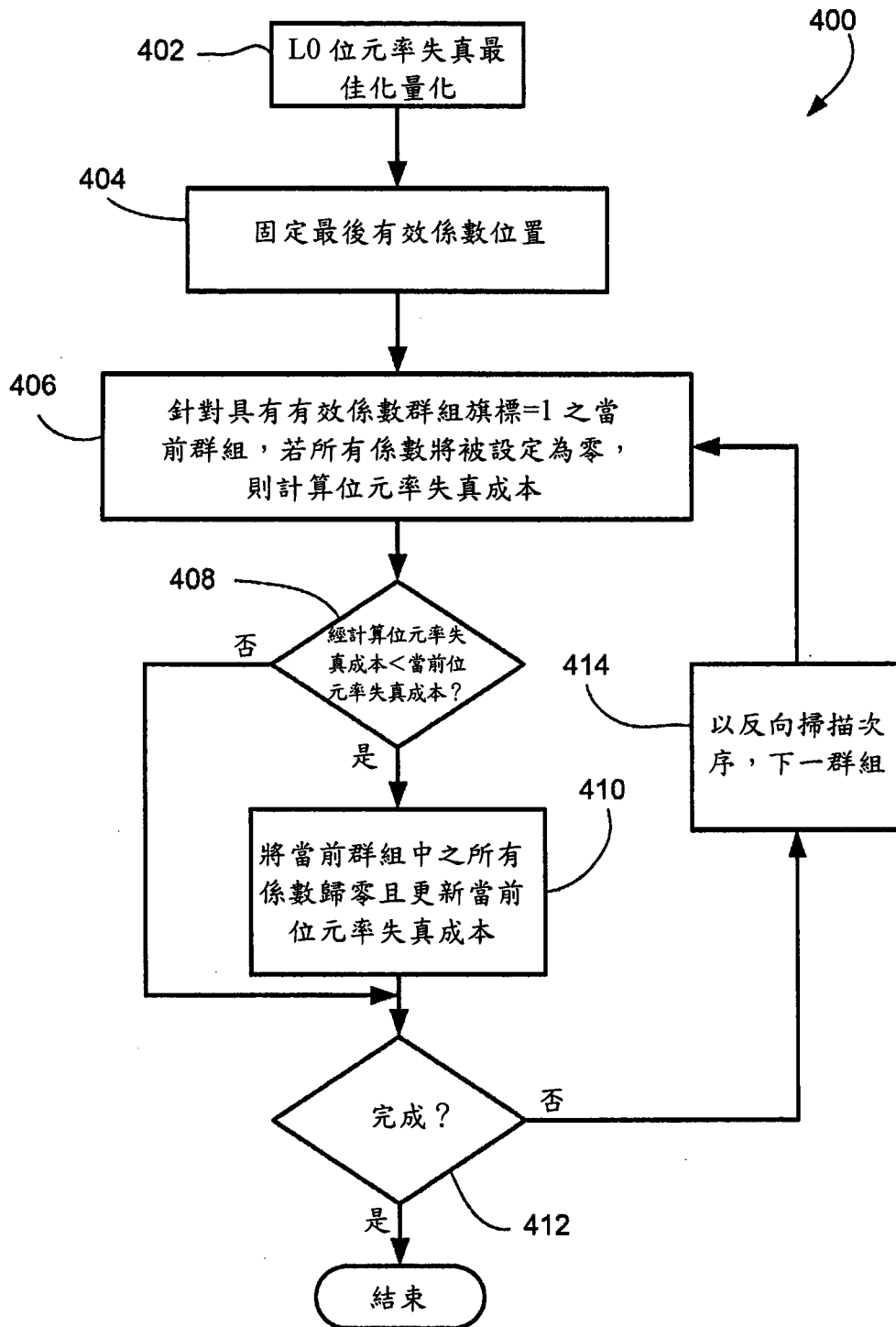


圖 9

xC

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	0	1	1	0	0	0	1	1	1	0	0	0	0	0
1	1	0	1	1	1	0	1	1	0	1	1	0	0	0	0	0
2	0	1	0	0	1	0	1	1	0	1	1	1	0	0	0	0
3	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0	0
4	1	1	0	0	1	0	1	0	0	1	0	1	0	0	0	0
5	1	1	0	1	0	0	1	0	1	0	0	1	0	0	0	0
6	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
7	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
8	0	1	1	1	0	1	0	1	1	1	1	0	0	1	0	0
9	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0
10	1	0	0	1	0	0	0	0	1	1	0	0	1	0	0	0
11	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

500

圖 10

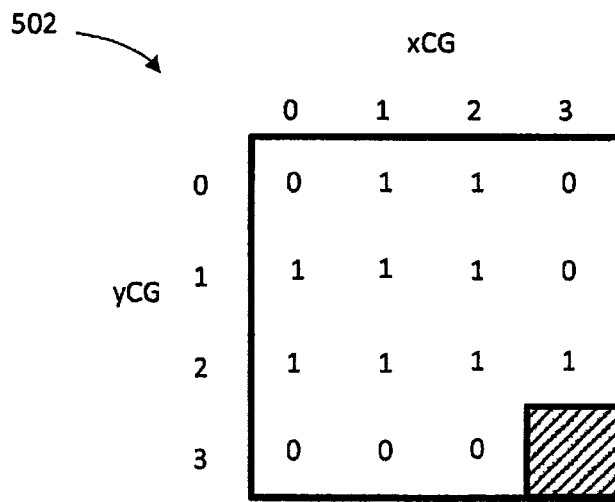


圖 11

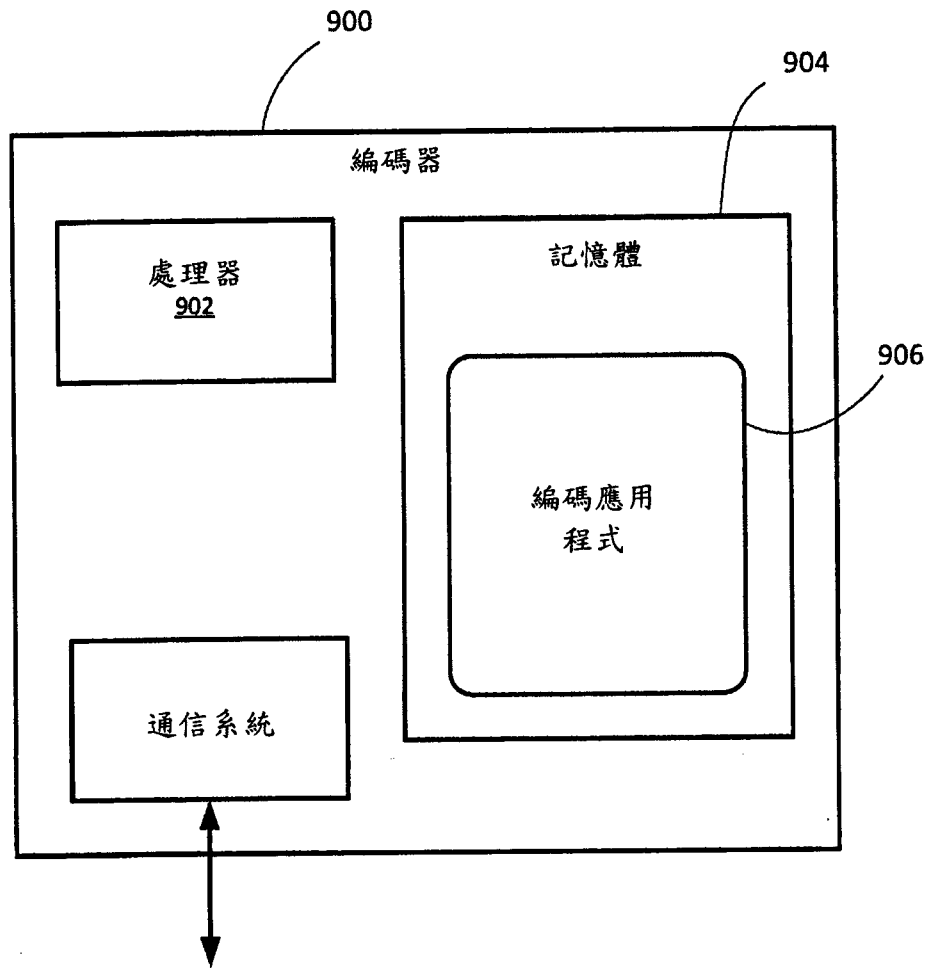


圖 12

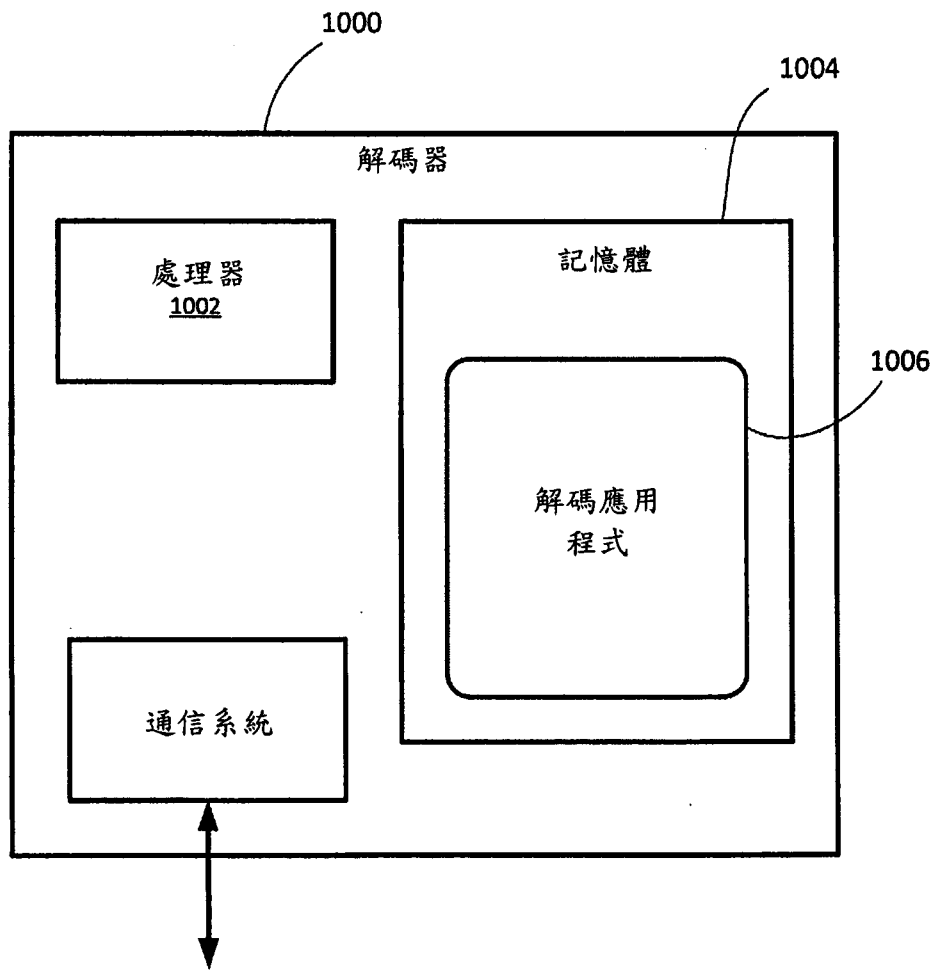


圖 13