

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4069486号
(P4069486)

(45) 発行日 平成20年4月2日(2008.4.2)

(24) 登録日 平成20年1月25日(2008.1.25)

(51) Int.Cl. F 1
G06T 15/00 (2006.01) G06T 15/00 100A
 G06T 15/00 400

請求項の数 8 (全 24 頁)

(21) 出願番号 特願平10-67242
 (22) 出願日 平成10年3月17日(1998.3.17)
 (65) 公開番号 特開平11-265459
 (43) 公開日 平成11年9月28日(1999.9.28)
 審査請求日 平成17年2月17日(2005.2.17)

(73) 特許権者 000002185
 ソニー株式会社
 東京都港区港南1丁目7番1号
 (74) 代理人 100094053
 弁理士 佐藤 隆久
 (72) 発明者 伊藤 豪
 東京都品川区東五反田1丁目14番10号
 株式会社ソニー木原研究所内

審査官 伊知地 和之

(56) 参考文献 特開平10-011594 (JP, A)
 特開平10-261095 (JP, A)
 特開平09-288742 (JP, A)

最終頁に続く

(54) 【発明の名称】 記憶回路制御装置およびグラフィック演算装置

(57) 【特許請求の範囲】

【請求項1】

マトリクス状に配置された複数の画素の色データを示す2次元画像データを記憶回路に記憶し、前記複数の画素の2次元配置に対応した2次元アドレス(U, V)を用いて前記記憶回路に記憶された前記2次元画像データにアクセスを行う記憶回路制御装置において

n (nは1以上の整数)ビットで表現された前記2次元アドレス(U, V)のUアドレスと、m (mは1以上の整数)ビットで表現された前記2次元アドレス(U, V)の前記Vアドレスとのそれぞれを構成するビットデータを組み合わせて、(n+m)ビットの1次元アドレスを生成するアドレス生成手段と、

前記生成された1次元アドレスを用いて、前記記憶回路にアクセスを行うデータアクセス手段と

を有し、

前記整数mが、(n-1)である場合に、

kを、 $0 < k < (n - 1)$ の整数とし、

前記Uアドレスを(U[n-1], ..., U[k], ..., U[0])のnビットで表現し、

前記Vアドレスを(V[n-2], ..., V[k], ..., V[0])の(n-1)ビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータU[n-1], ..., U[

k), ..., $U[0]$ と、前記Vアドレスの各ビットデータ $V[n-2]$, ..., $V[k]$, ..., $V[0]$ とを、組み合わせて、 $(2n-1)$ ビットの1次元アドレス($U[n-1]$, $V[n-2]$, $U[n-2]$, ..., $V[k]$, $U[k]$, ..., $V[0]$, $U[0]$)を生成する

記憶回路制御装置。

【請求項2】

立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算装置において、

10

複数のテクスチャデータを記憶した記憶回路と、

前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は1以上の整数)ビットで表現されたUアドレスと、 m (m は1以上の整数)ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成する2次元アドレス生成手段と、

前記2次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n+m)$ ビットの1次元アドレスを生成する1次元アドレス生成手段と、

前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段と

を有し、

20

前記整数 n と前記整数 m とが等しく、

k を、 $0 < k < (n-1)$ の整数とし、

前記Uアドレスを $(U[n-1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記Vアドレスを $(V[n-1], \dots, V[k], \dots, V[0])$ の n ビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータ $U[n-1]$, ..., $U[k]$, ..., $U[0]$ と、前記Vアドレスの各ビットデータ $V[n-1]$, ..., $V[k]$, ..., $V[0]$ とを、組み合わせて、 $2n$ ビットの1次元アドレス $(V[n-1], U[n-1], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

30

グラフィック演算装置。

【請求項3】

立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算装置において、

複数のテクスチャデータを記憶した記憶回路と、

前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は1以上の整数)ビットで表現されたUアドレスと、 m (m は1以上の整数)ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成する2次元アドレス生成手段と、

40

前記2次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n+m)$ ビットの1次元アドレスを生成する1次元アドレス生成手段と、

前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段と

を有し、

前記整数 m が、 $(n-1)$ である場合に、

k を、 $0 < k < (n-1)$ の整数とし、

前記Uアドレスを $(U[n-1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

50

前記Vアドレスを $(V[n-2], \dots, V[k], \dots, V[0])$ の $(n-1)$ ビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータ $U[n-1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n-2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n-1)$ ビットの1次元アドレス $(U[n-1], V[n-2], U[n-2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算装置。

【請求項4】

ディスプレイに表示する形状を表現する基本単位となる単位図形に張り合わせるイメージデータであるテクスチャデータを記憶する記憶回路と、

前記単位図形の頂点について、3次元座標 (x, y, z) 、R(赤)、G(緑)、B(青)データ、同次座標 (s, t) および同次項 q を含むポリゴンレンダリングデータを生成するポリゴンレンダリングデータ生成手段と、

前記単位図形の頂点のポリゴンレンダリングデータを補間して、前記単位図形内に位置する画素の補間データを生成する補間データ生成手段と、

前記補間データに含まれる前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は1以上の整数)ビットで表現されたUアドレスと、 m (m は1以上の整数)ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成する2次元アドレス生成手段と、

前記2次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n+m)$ ビットの1次元アドレスを生成する1次元アドレス生成手段と、

前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段と

を有し、

前記整数 n と前記整数 m とが等しく、

k を、 $0 < k < (n-1)$ の整数とし、

前記Uアドレスを $(U[n-1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記Vアドレスを $(V[n-1], \dots, V[k], \dots, V[0])$ の n ビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータ $U[n-1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n-1], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $2n$ ビットの1次元アドレス $(V[n-1], U[n-1], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算装置。

【請求項5】

ディスプレイに表示する形状を表現する基本単位となる単位図形に張り合わせるイメージデータであるテクスチャデータを記憶する記憶回路と、

前記単位図形の頂点について、3次元座標 (x, y, z) 、R(赤)、G(緑)、B(青)データ、同次座標 (s, t) および同次項 q を含むポリゴンレンダリングデータを生成するポリゴンレンダリングデータ生成手段と、

前記単位図形の頂点のポリゴンレンダリングデータを補間して、前記単位図形内に位置する画素の補間データを生成する補間データ生成手段と、

前記補間データに含まれる前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は1以上の整数)ビットで表現されたUアドレスと、 m (m は1以上の整数)ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成する2次元アドレス生成手段と、

前記2次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n+m)$ ビットの1次元アドレスを生成する1次元アドレス生成手段と、

10

20

30

40

50

前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段と

を有し、

前記整数 m が、 $(n - 1)$ である場合に、

k を、 $0 < k < (n - 1)$ の整数とし、

前記Uアドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記Vアドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの1次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算装置。

【請求項6】

マトリクス状に配置された複数の画素の色データを示す2次元画像データを記憶回路に記憶し、前記複数の画素の2次元配置に対応した2次元アドレス (U, V) を用いて前記記憶回路に記憶された前記2次元画像データにアクセスを行う記憶回路制御方法において

n (n は1以上の整数) ビットで表現された前記2次元アドレス (U, V) のUアドレスと、 m (m は1以上の整数) ビットで表現された前記2次元アドレス (U, V) の前記Vアドレスとのそれぞれを構成するビットデータを組み合わせて、 $(n + m)$ ビットの1次元アドレスを生成し、

前記生成された1次元アドレスを用いて、前記記憶回路にアクセスを行う

記憶回路制御方法であって、

前記整数 m が、 $(n - 1)$ である場合に、

k を、 $0 < k < (n - 1)$ の整数とし、

前記Uアドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記Vアドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビットで表現した場合に、

前記Uアドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの1次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

記憶回路制御方法。

【請求項7】

立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算方法において、

複数のテクスチャデータを記憶回路に記憶し、

前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s / q, t / q)$ に基づいて、 n (n は1以上の整数) ビットで表現されたUアドレスと、 m (m は1以上の整数) ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成し、

前記2次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの1次元アドレスを生成し、

前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付ける

10

20

30

40

50

グラフィック演算方法であって、
前記整数 n と前記整数 m とが等しく、
 k を、 $0 < k < (n - 1)$ の整数とし、
前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記 V アドレスを $(V[n - 1], \dots, V[k], \dots, V[0])$ の n ビットで表現した場合に、

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 1], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $2n$ ビットの 1 次元アドレス $(V[n - 1], U[n - 1], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する
グラフィック演算方法。

10

【請求項 8】

立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算方法において、

複数のテクスチャデータを記憶回路に記憶し、

前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスとからなる 2 次元アドレス (U, V) を生成し、

20

前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成し、

前記生成された 1 次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付ける

グラフィック演算方法であって、

前記整数 m が、 $(n - 1)$ である場合に、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

30

前記 V アドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビットで表現した場合に、

前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの 1 次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算方法。

【発明の詳細な説明】

【0001】

【発明が属する技術分野】

40

本発明は、例えば、テクスチャデータを記憶した記憶回路の記憶領域を効率的に使用できる記憶回路制御装置およびその方法と、グラフィック演算装置およびその方法とに関する。

【0002】

【従来の技術】

種々の CAD (Computer Aided Design) システムや、アミューズメント装置などにおいて、コンピュータグラフィックスがしばしば用いられている。特に、近年の画像処理技術の進展に伴い、3次元コンピュータグラフィックスを用いたシステムが急速に普及している。

このような 3次元コンピュータグラフィックスでは、マトリクス状に画素 (ピクセル) を

50

配置したCRT (Cathode Ray Tube)などのディスプレイに表示を行なうとき、レンダリング(Rendering) 処理を行なう。

このレンダリング処理は、各画素の色データを計算し、得られた色データを、当該画素に対応するディスプレイバッファ(フレームバッファ)に書き込む。レンダリング処理の手法の一つに、ポリゴン(Polygon)レンダリングがある。この手法では、立体モデルを三角形の単位図形(ポリゴン)の組み合わせとして表現しておき、このポリゴンを単位として描画を行なうことで、表示画面の色を決定する。

【0003】

このようなポリゴンレンダリングを用いた3次元コンピュータグラフィックシステムでは、描画時に、テクスチャマッピング処理が行なわれる。このテクスチャマッピング処理は、
10
三角形を単位として、イメージパターンを示すテクスチャデータをテクスチャバッファから読み出し、この読み出したテクスチャデータを立体モデルの表面に張り付け、リアリティの高い画像データを得るためのものである。

【0004】

このテクスチャマッピング処理では、以下に示すように、イメージデータに応じたイメージを映し出す画素を特定する2次元のテクスチャアドレスを算出し、これをテクスチャアドレスを用いて、テクスチャバッファに記憶されたテクスチャデータを参照する。

具体的には、まず、三角形の各頂点の同次座標(s, t)および同次項 q を示す(s_1, t_1, q_1), (s_2, t_2, q_2), (s_3, t_3, q_3)から、三角形の内部の各画素の(s, t, q)を線形補間して求める。
20

ここで、同次項 q は、簡単にいうと、拡大縮小率を示している。

【0005】

次に、各画素について、除算により、($s/q, t/q$)を算出し、 s/q および t/q のそれぞれにテクスチャサイズUSIZEおよびVSIZEを乗じてテクスチャ座標データ(u, v)を生成する。

次に、テクスチャ座標データ(u, v)を、テクスチャバッファ上のテクスチャアドレス(U, V)に変換し、このテクスチャアドレス(U, V)を用いて、テクスチャバッファからテクスチャデータを読み出す。

【0006】

上述した3次元コンピュータグラフィックシステムでは、テクスチャバッファをテクスチャアドレス(U, V)を用いて直接参照ができるように、テクスチャバッファの記憶領域に、テクスチャデータを U, V 座標系に対応する2次元的な配置で記憶する場合がある。
30
すなわち、2次元のテクスチャアドレス(U, V)を直接用いて、テクスチャバッファに記憶されたテクスチャデータにアクセスすることがある。この方法によれば、テクスチャデータにアクセスを行なう際の処理を簡単化できる。

しかしながら、この方法では、複数の種類のテクスチャデータをテクスチャバッファに記憶する場合に、記憶しようとするテクスチャデータのサイズと空き領域のサイズとの関係で、図10に示すように、有効に活用できない空き領域が生じ、記憶領域を効率的に利用できないという問題がある。

【0007】

例えば、図10に示すように、 U, V 方向のアドレス長が異なる複数のテクスチャデータ400, 401, 402, 403, 406を、テクスチャアドレス(U, V)によって直接参照できるようにテクスチャバッファのアドレス空間に記憶すると、記憶しようとするテクスチャデータの2次元的なサイズと空き領域の2次元的なサイズとの関係で、テクスチャデータを記憶できない空き領域410, 411が生じてしまう。
40

【0008】

その結果、記憶するテクスチャデータのデータ量に比べて、非常に大きな記憶容量を持つテクスチャバッファを用いる必要があり、システムが大規模化および高価格化するという問題がある。

【0009】

そのため、従来では、テクスチャバッファの記憶領域を効率的に利用するために、「物理アドレス $A = V \times (\text{テクスチャの幅}) + U$ 」に基づいて、2次元のテクスチャアドレス (U, V) から1次元の物理アドレス A を算出し、この物理アドレス A を用いて、テクスチャバッファにアクセスを行なっている。このようにすることで、図11に示すように、テクスチャバッファの記憶領域に空き領域をつくることなく、テクスチャデータを記憶できる。

なお、「テクスチャの幅」は、テクスチャバッファのアドレス空間における、 U 方向のアドレス長を示している。

【0010】

図12は、従来の3次元コンピュータグラフィックシステムの部分構成図である。

図12に示すように、テクスチャマッピング装置101に内蔵されたアドレス変換装置104において、上述したように、三角形の頂点の (s_1, t_1, q_1) , (s_2, t_2, q_2) , (s_3, t_3, q_3) から、各画素の物理アドレス A が算出される。そして、当該算出された物理アドレス A を用いて、テクスチャバッファ102からテクスチャマッピング装置101にテクスチャデータ (R, G, B, \dots) が読み出され、このテクスチャデータ (R, G, B, \dots) が立体モデルの表面に対応する画素に張り付けられ、描画データ $S101$ が生成される。この描画データ $S101$ は、ディスプレイバッファ103に書き込まれる。

【0011】

また、高速な3次元コンピュータグラフィックシステムでは、例えば、図13に示すように、それぞれアドレス変換装置 $104_1 \sim 104_n$ を内蔵した n 個のテクスチャマッピング装置 $101_1 \sim 101_n$ を備え、 n 個の画素について、テクスチャマッピング処理が同時に並行して行なわれ、描画データ $S101_1 \sim S101_n$ がディスプレイバッファに同時に書き込まれる。

【0012】

【発明が解決しようとする課題】

しかしながら、上述した3次元コンピュータグラフィックシステムでは、上述したように「物理アドレス $A = V \times (\text{テクスチャの幅}) + U$ 」を用いて、2次元のテクスチャアドレス (U, V) から、1次元の物理アドレス A を生成するとき、「テクスチャの幅」に応じた乗算を行なう大規模な乗算回路が必要になる。その結果、システムが大規模化してしまふという問題がある。

特に、図13に示すような、複数のテクスチャマッピング装置 $101_1 \sim 101_n$ を備えた場合には、回路規模の問題は深刻になる。

【0013】

本発明は上述した従来技術の問題点に鑑みてなされ、小規模な回路構成で、テクスチャバッファの記憶領域を効率的に使用できる記憶回路制御装置およびグラフィック演算装置を提供することを目的とする。

また、本発明は、テクスチャバッファの記憶領域を効率的に使用できる記憶回路制御方法およびグラフィック演算方法を提供することを目的とする。

【0014】

【課題を解決するための手段】

本発明によれば、マトリクス状に配置された複数の画素の色データを示す2次元画像データを記憶回路に記憶し、前記複数の画素の2次元配置に対応した2次元アドレス (U, V) を用いて前記記憶回路に記憶された前記2次元画像データにアクセスを行う記憶回路制御装置において、

n (n は1以上の整数) ビットで表現された前記2次元アドレス (U, V) の U アドレスと、 m (m は1以上の整数) ビットで表現された前記2次元アドレス (U, V) の前記 V アドレスとのそれぞれを構成するビットデータを組み合わせ、 $(n+m)$ ビットの1次元アドレスを生成するアドレス生成手段と、前記生成された1次元アドレスを用いて、前記記憶回路にアクセスを行うデータアクセス手段とを有し、

10

20

30

40

50

前記整数 m が、 $(n - 1)$ である場合に、
 k を、 $0 < k < (n - 1)$ の整数とし、
 前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表
 現し、

前記 V アドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビ
 ットで表現した場合に、

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの 1 次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

10

記憶回路制御装置が提供される。

【0015】

また本発明によれば、立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算装置において、

複数のテクスチャデータを記憶した記憶回路と、前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスと
 からなる 2 次元アドレス (U, V) を生成する 2 次元アドレス生成手段と、前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成する 1 次元アドレス生成手段と、前記生成された 1 次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段とを有し、

20

前記整数 n と前記整数 m とが等しく、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記 V アドレスを $(V[n - 1], \dots, V[k], \dots, V[0])$ の n ビットで表現した場合に、

30

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 1], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $2n$ ビットの 1 次元アドレス $(V[n - 1], U[n - 1], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算装置が提供される。

【0016】

本発明によれば、立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算装置において、

40

複数のテクスチャデータを記憶した記憶回路と、前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスと
 からなる 2 次元アドレス (U, V) を生成する 2 次元アドレス生成手段と、前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成する 1 次元アドレス生成手段と、前記生成された 1 次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段とを有し、

前記整数 m が、 $(n - 1)$ である場合に、

50

k を、 $0 < k < (n - 1)$ の整数とし、
前記 U アドレスを ($U[n - 1], \dots, U[k], \dots, U[0]$) の n ビットで表現し、

前記 V アドレスを ($V[n - 2], \dots, V[k], \dots, V[0]$) の $(n - 1)$ ビットで表現した場合に、

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの 1 次元アドレス ($U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0]$) を生成する

10

グラフィック演算装置が提供される。

【0017】

本発明によれば、ディスプレイに表示する形状を表現する基本単位となる単位図形に張り合わせるイメージデータであるテクスチャデータを記憶する記憶回路と、

前記単位図形の頂点について、3次元座標 (x, y, z) 、R (赤)、G (緑)、B (青) データ、同次座標 (s, t) および同次項 q を含むポリゴンレンダリングデータを生成するポリゴンレンダリングデータ生成手段と、前記単位図形の頂点のポリゴンレンダリングデータを補間して、前記単位図形内に位置する画素の補間データを生成する補間データ生成手段と、前記補間データに含まれる前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスとからなる 2 次元アドレス (U, V) を生成する 2 次元アドレス生成手段と、前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成する 1 次元アドレス生成手段と、前記生成された 1 次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段とを有し、

20

前記整数 n と前記整数 m とが等しく、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを ($U[n - 1], \dots, U[k], \dots, U[0]$) の n ビットで表現し、

30

前記 V アドレスを ($V[n - 1], \dots, V[k], \dots, V[0]$) の n ビットで表現した場合に、

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 1], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $2n$ ビットの 1 次元アドレス ($V[n - 1], U[n - 1], \dots, V[k], U[k], \dots, V[0], U[0]$) を生成する

グラフィック演算装置が提供される。

【0018】

また本発明によれば、ディスプレイに表示する形状を表現する基本単位となる単位図形に張り合わせるイメージデータであるテクスチャデータを記憶する記憶回路と、前記単位図形の頂点について、3次元座標 (x, y, z) 、R (赤)、G (緑)、B (青) データ、同次座標 (s, t) および同次項 q を含むポリゴンレンダリングデータを生成するポリゴンレンダリングデータ生成手段と、前記単位図形の頂点のポリゴンレンダリングデータを補間して、前記単位図形内に位置する画素の補間データを生成する補間データ生成手段と、前記補間データに含まれる前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s/q, t/q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスとからなる 2 次元アドレス (U, V) を生成する 2 次元アドレス生成手段と、前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成する 1 次元アドレス生成手段と、前記生成された 1 次元アドレスを用いて、前記記憶回路から前記

40

50

テクスチャデータを読み出し、前記単位図形に張り付けるデータ読み出し手段とを有し、
前記整数 m が、 $(n - 1)$ である場合に、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記 V アドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビットで表現した場合に、

前記アドレス生成手段は、前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの 1 次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算装置が提供される。

【0019】

本発明によれば、マトリクス状に配置された複数の画素の色データを示す 2 次元画像データを記憶回路に記憶し、前記複数の画素の 2 次元配置に対応した 2 次元アドレス (U, V) を用いて前記記憶回路に記憶された前記 2 次元画像データにアクセスを行う記憶回路制御方法において、

n (n は 1 以上の整数) ビットで表現された前記 2 次元アドレス (U, V) の U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 2 次元アドレス (U, V) の前記 V アドレスとのそれぞれを構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成し、前記生成された 1 次元アドレスを用いて、前記記憶回路にアクセスを行う記憶回路制御方法であって、

前記整数 m が、 $(n - 1)$ である場合に、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

前記 V アドレスを $(V[n - 2], \dots, V[k], \dots, V[0])$ の $(n - 1)$ ビットで表現した場合に、

前記 U アドレスの各ビットデータ $U[n - 1], \dots, U[k], \dots, U[0]$ と、前記 V アドレスの各ビットデータ $V[n - 2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n - 1)$ ビットの 1 次元アドレス $(U[n - 1], V[n - 2], U[n - 2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

記憶回路制御方法が提供される。

【0020】

本発明によれば、立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項 q に応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算方法において、

複数のテクスチャデータを記憶回路に記憶し、前記同次座標 (s, t) を前記同次項 q で除算した除算結果 $(s / q, t / q)$ に基づいて、 n (n は 1 以上の整数) ビットで表現された U アドレスと、 m (m は 1 以上の整数) ビットで表現された前記 V アドレスとからなる 2 次元アドレス (U, V) を生成し、前記 2 次元アドレスの U, V を構成するビットデータを組み合わせて、 $(n + m)$ ビットの 1 次元アドレスを生成し、前記生成された 1 次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるグラフィック演算方法であって、

前記整数 n と前記整数 m とが等しく、

k を、 $0 < k < (n - 1)$ の整数とし、

前記 U アドレスを $(U[n - 1], \dots, U[k], \dots, U[0])$ の n ビットで表現し、

10

20

30

40

50

前記Vアドレスを $(V[n-1], \dots, V[k], \dots, V[0])$ のnビットで表現した場合に、

前記アドレス生成手段は、前記Uアドレスの各ビットデータ $U[n-1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n-1], \dots, V[k], \dots, V[0]$ とを、組み合わせて、2nビットの1次元アドレス $(V[n-1], U[n-1], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成するグラフィック演算方法が提供される。

【0021】

また本発明によれば、立体モデルを複数の単位図形の組み合わせで表現し、前記単位図形の内部に位置する各画素の画像データに含まれる同次座標 (s, t) および同次項qに
10 応じたアドレスを用いて、前記単位図形に張り付ける画像データであるテクスチャデータを記憶回路から読み出して単位図形に張り付けるグラフィック演算方法において、

複数のテクスチャデータを記憶回路に記憶し、前記同次座標 (s, t) を前記同次項qで除算した除算結果 $(s/q, t/q)$ に基づいて、n(nは1以上の整数)ビットで表現されたUアドレスと、m(mは1以上の整数)ビットで表現された前記Vアドレスとからなる2次元アドレス (U, V) を生成し、前記2次元アドレスのU, Vを構成するビットデータを組み合わせて、 $(n+m)$ ビットの1次元アドレスを生成し、前記生成された1次元アドレスを用いて、前記記憶回路から前記テクスチャデータを読み出し、前記単位図形に張り付けるグラフィック演算方法であって、

前記整数mが、 $(n-1)$ である場合に、
20

kを、 $0 < k < (n-1)$ の整数とし、

前記Uアドレスを $(U[n-1], \dots, U[k], \dots, U[0])$ のnビットで表現し、

前記Vアドレスを $(V[n-2], \dots, V[k], \dots, V[0])$ の $(n-1)$ ビットで表現した場合に、

前記Uアドレスの各ビットデータ $U[n-1], \dots, U[k], \dots, U[0]$ と、前記Vアドレスの各ビットデータ $V[n-2], \dots, V[k], \dots, V[0]$ とを、組み合わせて、 $(2n-1)$ ビットの1次元アドレス $(U[n-1], V[n-2], U[n-2], \dots, V[k], U[k], \dots, V[0], U[0])$ を生成する

グラフィック演算方法が提供される。
30

【0026】

【発明の実施の形態】

以下、本実施形態においては、家庭用ゲーム機などに適用される、任意の3次元物体モデルに対する所望の3次元画像をCRTなどのディスプレイ上に高速に表示する3次元コンピュータグラフィックシステムについて説明する。

図1は、本実施形態の3次元コンピュータグラフィックシステム1のシステム構成図である。

3次元コンピュータグラフィックシステム1は、立体モデルを単位図形である三角形(ポリゴン)の組み合わせとして表現し、このポリゴンを描画することで表示画面の各画素の色を決定し、ディスプレイに表示するポリゴンレンダリング処理を行うシステムである。
40 また、3次元コンピュータグラフィックシステム1では、平面上の位置を表現する (x, y) 座標の他に、奥行きを表すz座標を用いて3次元モデルを表し、この (x, y, z) の3つの座標で3次元空間の任意の一点を特定する。

【0027】

図1に示すように、3次元コンピュータグラフィックシステム1は、メインメモリ2、I/Oインタフェース回路3、メインプロセッサ4およびレンダリング回路5が、メインバス6を介して接続されている。

以下、各構成要素の機能について説明する。

メインプロセッサ4は、例えば、ゲームの進行状況などに応じて、メインメモリ2から必要なグラフィックデータを読み出し、このグラフィックデータに対してクリッピング(Cli
50

pping)処理、ライティング(Lighting)処理およびジオメトリ(Geometry)処理などを行い、ポリゴンレンダリングデータを生成する。メインプロセッサ4は、ポリゴンレンダリングデータS4を、メインバス6を介してレンダリング回路5に出力する。

I/Oインタフェース回路3は、必要に応じて、外部からポリゴンレンダリングデータを入力し、これをメインバス6を介してレンダリング回路5に出力する。

【0028】

ここで、ポリゴンレンダリングデータは、ポリゴンの各3頂点の $(x, y, z, R, G, B, s, t, q, F)$ のデータを含んでいる。

ここで、 (x, y, z) データは、ポリゴンの頂点の3次元座標を示し、 (R, G, B) データは、それぞれ当該3次元座標における赤、緑、青の輝度値を示している。

データは、これから描画する画素と、ディスプレイバッファ21に既に記憶されている画素との R, G, B データのブレンド(混合)係数を示している。

(s, t, q) データのうち、 (s, t) は、対応するテクスチャの同次座標を示しており、 q は同次項を示している。ここで、「 s/q 」および「 t/q 」に、それぞれテクスチャサイズ $U SIZE$ および $V SIZE$ を乗じてテクスチャ座標データ (u, v) が得られる。テクスチャバッファ20に記憶されたテクスチャデータへのアクセスは、テクスチャ座標データ (u, v) を用いて行われる。

Fデータは、フォグの値を示している。

すなわち、ポリゴンレンダリングデータは、三角形の各頂点の物理座標値と、それぞれの頂点の色とテクスチャおよびフォグの値のデータを示している。

【0029】

以下、レンダリング回路5について詳細に説明する。

図1に示すように、レンダリング回路5は、DDA(Digital Differential Analyzer)セットアップ回路10、補間データ生成手段としてのトライアングルDDA回路11、テクスチャエンジン回路12、メモリI/F回路13、CRTコントローラ回路14、RAMDAC回路15、DRAM16およびSRAM17を有する。

DRAM16は、記憶回路としてのテクスチャバッファ20、ディスプレイバッファ21、zバッファ22およびテクスチャCLUTバッファ23として機能する。

【0030】

DDAセットアップ回路10

DDAセットアップ回路10は、後段のトライアングルDDA回路11において物理座標系上の三角形の各頂点の値を線形補間して、三角形の内部の各画素の色と深さ情報を求めるに先立ち、ポリゴンレンダリングデータS4が示す (z, R, G, B, s, t, q, F) データについて、三角形の辺と水平方向の差分などを求めるセットアップ演算を行う。

このセットアップ演算は、具体的には、開始点の値と終点の値と、開始点と終点との距離を用いて、単位長さ移動した場合における、求めようとしている値の変分を算出する。

【0031】

DDAセットアップ回路10は、算出した変分データS10をトライアングルDDA回路11に出力する。

【0032】

トライアングルDDA回路11

トライアングルDDA回路11は、DDAセットアップ回路10から入力した変分データS10を用いて、三角形内部の各画素における線形補間された (z, R, G, B, s, t, q, F) データを算出する。

トライアングルDDA回路11は、各画素の (x, y) データと、当該 (x, y) 座標における (z, R, G, B, s, t, q, F) データとを、DDAデータS11としてテクスチャエンジン回路12に出力する。

本実施形態では、トライアングルDDA回路11は、並行して処理を行う矩形内に位置する8($= 2 \times 4$)画素分を単位として、DDAデータS11をテクスチャエンジン回路1

10

20

30

40

50

2 に出力する。

【 0 0 3 3 】

テクスチャエンジン回路 1 2

テクスチャエンジン回路 1 2 は、テクスチャデータの縮小率の選択処理、「 s/q 」および「 t/q 」の算出処理、テクスチャ座標データ (u, v) の算出処理、2次元のテクスチャアドレス (U, V) の算出処理、1次元の物理アドレス A の生成、テクスチャバッファ 20 からの (R, G, B, t) データの読み出し処理、および、混合処理 (テクスチャブレンディング処理) を順にパイプライン方式で行う。

このとき、テクスチャエンジン回路 1 2 は、所定の矩形領域内に位置する 8 画素についての処理を同時に並行して行う。

10

【 0 0 3 4 】

図 2 は、テクスチャエンジン回路 1 2 の構成図である。

図 2 に示すように、テクスチャエンジン回路 1 2 は、縮小率演算回路 3 0 4、テクスチャデータ読み出し回路 3 0 5 およびテクスチャブレンド回路 3 0 6 を有する。

【 0 0 3 5 】

縮小率演算回路 3 0 4 は、DDA データ S 1 1 に含まれる 8 画素分の (s, t, q) データ S 1 1 a₁ ~ S 1 1 a₈ などを用いて、テクスチャデータの縮小率 lod を算出する。ここで、縮小率は、元画像のテクスチャデータを、どの程度縮小したものであるかを示すものであり、元画像の縮小率を $1/1$ とした場合には、 $1/2, 1/4, 1/8, \dots$ となる。

20

【 0 0 3 6 】

テクスチャバッファ 20 には、例えば、図 3 に示すように、 $lod = 0, 1, 2, 3, 4$ のテクスチャデータ 3 2 0, 3 2 1, 3 2 2, 3 2 3 が記憶されている。

なお、テクスチャバッファ 20 の記憶領域のアドレス空間は、図 3 に示すように、 U, V 座標系で表現され、複数の縮小率に対応したテクスチャデータが記憶されている記憶領域の基準アドレス (開始アドレス) は、縮小率 lod に基づいて算出される。図 7 に示す例では、テクスチャデータ 3 2 0, 3 2 1, 3 2 2, 3 2 3 の基準アドレスは、($ubase_0, vbase_0$), ($ubase_1, vbase_1$), ($ubase_2, vbase_2$), ($ubase_3, vbase_3$) となる。

また、テクスチャバッファ 20 に記憶されているテクスチャデータにおける各画素についてのテクスチャアドレス (U, V) は、基準アドレス ($ubase, vbase$) と、テクスチャ座標データ (u, v) とを加算したアドレスとなる。

30

【 0 0 3 7 】

〔テクスチャデータ読み出し回路 3 0 5〕

テクスチャデータ読み出し回路 3 0 5 は、DDA データ S 1 1 に含まれる 8 画素分の (s, t, q) データ S 1 1 a₁ ~ S 1 1 a₈ と、縮小率演算回路 3 0 4 からの縮小率 lod と、テクスチャサイズ $USIZE$ および $VSIZE$ とを入力し、8 画素のそれぞれに対応した、テクスチャデータ S 1 7₁ ~ S 1 7₈ をテクスチャバッファ 20 から読み出し、これをテクスチャブレンド回路 3 0 6 に出力する。

【 0 0 3 8 】

40

図 4 はテクスチャデータ読み出し回路 3 0 5 の構成図である。図 5 は、テクスチャデータ読み出し回路 3 0 5 における処理のフローチャートである。

図 4 に示すように、テクスチャデータ読み出し回路 3 0 5 は、 u, v 算出回路 5 0 1、 U, V 算出回路 5 0 2、アドレス生成手段としての物理アドレス生成回路 5 0 3 およびデータアクセス手段としてのアクセス制御回路 5 0 4 を有する。ステップ S 2 1: テクスチャデータ読み出し回路 3 0 5 では、まず、 u, v 算出回路 5 0 1 において、8 画素分の (s, t, q) データ S 1 1 a₁ ~ S 1 1 a₈ のそれぞれについて、 s データを q データで除算する演算と、 t データを q データで除算する演算とを行い、除算結果「 s/q 」および「 t/q 」を算出する。

そして、除算結果「 s/q 」および「 t/q 」に、それぞれテクスチャサイズ $USIZE$

50

および $VSIZE$ を乗じて、各画素に対応したテクスチャ座標データ $(u_1, v_1) \sim (u_8, v_8)$ を算出する。

【0039】

ステップ S22: U, V 算出回路 502 は、例えば、予め用意したアドレステーブルを参照して、縮小率 lod に対応する基準アドレス $(ubase, vbase)$ を得る。

そして、U, V 算出回路 502 は、基準アドレス $(ubase, vbase)$ と、u, v 算出回路 501 から入力したテクスチャ座標データ $(u_1, v_1) \sim (u_8, v_8)$ とを加算して、テクスチャバッファ 20 の記憶領域を 2 次元の UV 座標系で表した場合のテクスチャアドレス $(U_1, V_1) \sim (U_8, V_8)$ を生成する。

本実施形態では、「i」を「1 ~ 8」の整数とした場合に、テクスチャアドレス (U_i, V_i) の U_i および V_i は、それぞれ下記 (1) および (2) に示す 3 ビットからなる。

【0040】

【数 1】

$$U_i = \{ U_i [2], U_i [1], U_i [0] \} \quad \dots (1)$$

【0041】

【数 2】

$$V_i = \{ V_i [2], V_i [1], V_i [0] \} \quad \dots (2)$$

【0042】

ステップ S23: 物理アドレス生成回路 503 は、U, V 算出回路 502 から入力したテクスチャアドレス $(U_1, V_1) \sim (U_8, V_8)$ のそれぞれについて、 U_i および V_i を構成するビットを、下記 (3) に示すパターンで結合し、物理アドレス A を生成する。この物理アドレス A の生成は、簡単なビット入れ替え操作のみで実現されるため、小規模な回路構成で高速に行なうことができる。

【0043】

【数 3】

$$\text{物理アドレス } A = (V_i [2], U_i [2], V_i [1], U_i [1], V_i [0], U_i [0]) \quad \dots (3)$$

【0044】

上記 (3) に示す物理アドレス A の生成は、図 6 で示される。図 6 において、縦軸 V が 3 ビットで示される V_i の値を示し、横軸 U が 3 ビットで示される U_i の値を示し、マトリクス状に配置された「0」~「63」が物理アドレス A の値を示している。

すなわち、3 ビットの U_i および V_i で表されるマトリクス状に位置する 2 次元のテクスチャアドレス (U_i, V_i) は、上記式 (3) によって、「0」~「63」の連続した 1 次元の物理アドレス A に変換される。

但し、本発明では、 U_i は 2^n (n は 1 以上の整数) ビットであり、 V_i は 2^m (m は 1 以上の整数) ビット数であり、しかも、 $n = m$ 、あるいは、 $m = n - 1$ である。

【0045】

上記式 (3) の変換の具体例を、図 6 を参照していくつか例示する。

例えば、 $U_i = (0, 1, 0)$ 、 $V_i = (1, 0, 0)$ である場合には、上記式 (3) によって、物理アドレス $A = (1, 0, 0, 1, 0, 0)$ となり、10 進数で表すと、「36」となる。ここで、 $U_i = (0, 1, 0) = 2$ であり、 $V_i = (1, 0, 0) = 4$ であり、図 6 において、 $U = 2$ 、 $V = 4$ の位置 A は「36」になっている。

また、 $U_i = (0, 0, 1)$ 、 $V_i = (0, 1, 0)$ である場合には、上記式 (3) によって、物理アドレス $A = (0, 0, 1, 0, 0, 1)$ となり、10 進数で表すと、「9」となる。ここで、 $U_i = (0, 0, 1) = 1$ であり、 $V_i = (0, 1, 0) = 2$ であり、図 6 において、 $U = 1$ 、 $V = 2$ の位置 B は「9」になっている。

【0046】

ステップ S24: アクセス制御回路 504 は、物理アドレス生成回路 503 から入力した 1 次元の物理アドレス A を、図 1 に示すメモリ I/F 回路 13 を介して、テクスチャバッ

10

20

30

40

50

ファ20に出力し、テクスチャデータである (R, G, B, t) データ $S17_1 \sim S17_8$ を読み出す。

なお、SRAM17には、テクスチャバッファ20に記憶されているテクスチャデータのコピーが記憶されており、テクスチャエンジン回路12は、実際には、メモリI/F回路13を介してSRAM17に記憶されているテクスチャデータを読み出す。

【0047】

ステップS25：アクセス制御回路504は、ステップS24で読み出した (R, G, B, t) データ $S17_1 \sim S17_8$ をテクスチャブレンド回路306に出力する。

【0048】

〔テクスチャブレンド回路306〕

テクスチャブレンド回路306は、DDAデータ $S11$ に含まれる8画素分の (R, G, B) データ $S11b_1 \sim S11b_8$ と、テクスチャデータ読み出し回路305が読み出した (R, G, B, t) データ $S17_1 \sim S17_8$ とを入力し、それぞれ (R, G, B) データ $S11b_1 \sim S11b_8$ と、データ $S17_1 \sim S17_8$ に含まれる (R, G, B) データとを、データ $S17_1 \sim S17_8$ に含まれる t で示される混合値で混合し、 (R, G, B) データ $S306_1 \sim S306_8$ を生成する。

そして、DDAデータに含まれるデータ $S11d_1 \sim S11d_8$ と、 (R, G, B) データ $S306_1 \sim S306_8$ とが、 (R, G, B, t) データ $S12a_1 \sim S12a_8$ として、メモリI/F回路13に出力される。

【0049】

なお、テクスチャエンジン回路12は、フルカラー方式の場合には、テクスチャバッファ20から読み出した (R, G, B, t) データを直接用いる。一方、テクスチャエンジン回路12は、インデックスカラー方式の場合には、予め作成したカラーlookupテーブル(CLUT)をテクスチャCLUTバッファ23から読み出して、内蔵するSRAMに転送および記憶し、このカラーlookupテーブルを用いて、テクスチャバッファ20から読み出したカラーインデックスに対応する (R, G, B) データを得る。

【0050】

DRAM16およびSRAM17

図7は、DRAM16、SRAM17、および、メモリI/F回路13のDRAM16およびSRAM17へのアクセス機能を持つブロックの構成図である。図7に示すように、図1に示すDRAM16およびSRAM17は、メモリモジュール200、201、202、203を有する。

メモリモジュール200は、メモリ210、211を有する。

メモリ210は、DRAM16の一部を構成するバンク220₁、220₂と、SRAM17の一部を構成するバンク221₁、221₂とを有する。

また、メモリ211は、DRAM16の一部を構成するバンク220₁、220₂と、SRAM17の一部を構成するバンク221₁、221₂とを有する。

バンク220₁、220₂、221₁、221₂に対しては同時アクセスが可能である。

なお、メモリモジュール201、202、203は、基本的に、メモリモジュール200と同じ構成をしている。

【0051】

ここで、メモリモジュール200、201、202、203の各々は、図1に示すテクスチャバッファ20、ディスプレイバッファ21、Zバッファ22およびテクスチャCLUTバッファ23の全ての機能を持つ。

すなわち、メモリモジュール200、201、202、203の各々は、対応する画素のテクスチャデータ、描画データ (R, G, B) データ、zデータおよびテクスチャカラーlookupテーブルデータの全てを記憶する。

但し、メモリモジュール200、201、202、203は、相互で異なる画素についてのデータを記憶する。

ここで、同時に処理される16画素についてのテクスチャデータ、描画データ、zデータ

10

20

30

40

50

およびテクスチャカラーlookupテーブルデータが、相互に異なるバンク 210_1 , 210_2 , 211_1 , 211_2 , 212_1 , 212_2 , 213_1 , 213_2 , 214_1 , 214_2 , 215_1 , 215_2 , 216_1 , 216_2 , 217_1 , 217_2 に記憶される。

これにより、DRAM 16 に対して、16 画素についてのデータが同時にアクセス可能になる。

【0052】

なお、バンク 220_1 , 220_2 , 221_1 , 221_2 , 222_1 , 222_2 , 223_1 , 223_2 , 224_1 , 224_2 , 225_1 , 225_2 , 226_1 , 226_2 , 227_1 , 227_2 には、それぞれバンク 210_1 , 210_2 , 211_1 , 211_2 , 212_1 , 212_2 , 213_1 , 213_2 , 214_1 , 214_2 , 215_1 , 215_2 , 216_1 , 216_2 , 217_1 , 217_2 に記憶されたテクスチャデータのコピーが記憶されている。

10

【0053】

メモリ I / F 回路 13

また、メモリ I / F 回路 13 は、テクスチャエンジン回路 12 から入力した (R , G , B ,) データ $S12a_1 \sim S12a_8$ 、すなわち画素データ $S12a$ に対応する z データと、z バッファ 22 に記憶されている z データとの比較を行い、入力した画素データ $S12a$ によって描画される画像が、前回、ディスプレイバッファ 21 に書き込まれた画像より、手前 (視点側) に位置するか否かを判断し、手前に位置する場合には、画素データ $S12a$ に対応する z データで z バッファ 22 に記憶された z データを更新する。

20

また、メモリ I / F 回路 13 は、必要に応じて、画素データ $S12a$ に含まれる (R , G , B) データと、既にディスプレイバッファ 21 に記憶されている (R , G , B) データとを、画素データ $S12a$ に対応する データが示す混合値で混合する、いわゆる ブレンディング処理を行い、混合後の (R , G , B) データをディスプレイバッファ 21 に書き込む (打ち込む) 。

【0054】

メモリ I / F 回路 13 は、DRAM 16 に対して 16 画素について同時にアクセスを行なう。

図 7 に示すように、メモリ I / F 回路 13 は、メモリコントローラ 240 , 241 , 242 , 243、アドレスコンバータ 250 , 251 , 252 , 253、ディストリビュータ 260 および読み出しコントローラ 262 を有する。

30

【0055】

ディストリビュータ 260 は、例えば、書き込み時に、16 画素分の (R , G , B) データを入力し、これらを、各々 4 画素分のデータからなる 4 つの画像データ $S260_0$, $S260_1$, $S260_2$, $S260_3$ に分割し、それぞれをアドレスコンバータ 250 , 251 , 252 , 253 に出力する。

ここで、1 画素分の (R , G , B) データおよび z データは、それぞれ 32 ビットからなる。

【0056】

アドレスコンバータ 250 , 251 , 252 , 253 は、書き込み時に、ディストリビュータ 260 から入力した (R , G , B) データおよび z データに対応したアドレスを、それぞれメモリモジュール 200 , 201 , 202 , 203 内のアドレスに変換し、それぞれ変換したアドレス $S250$, $S251$, $S252$, $S253$ をメモリコントローラ 240 に出力する。

40

【0057】

メモリコントローラ 240 , 241 , 242 , 243 は、それぞれ配線群 270 , 271 , 272 , 273 を介してメモリモジュール 200 , 201 , 202 , 203 に接続されており、書き込み時にメモリモジュール 200 , 201 , 202 , 203 に対してのアクセスを制御する。

50

具体的には、メモリコントローラ240, 241, 242, 243は、ディストリビュータ260から入力した4画素分の(R, G, B)データおよびzデータを、配線群270, 271, 272, 273を介してメモリモジュール200, 201, 202, 203に同時に書き込む。

このとき、例えば、メモリモジュール200では、バンク210₁, 210₂, 210₃, 210₄の各々に、1画素分の(R, G, B)データおよびzデータが記憶される。メモリモジュール201, 202, 203についても同じである。

なお、本実施形態では、配線群270, 271, 272, 273の各々は、256ビットである。

【0058】

読み出しコントローラ262は、配線群280を介してメモリモジュール200, 201, 202, 203と接続されており、読み出し時に、メモリモジュール200, 201, 202, 203から、8画素あるいは16画素単位で、テクスチャデータ、(R, G, B)データ、zデータおよびテクスチャカラーlookupテーブルデータを配線群280を介して読み出す。

なお、本実施形態では、配線群280は、1024ビットである。

【0059】

CRTコントローラ回路14

CRTコントローラ回路14は、与えられた水平および垂直同期信号に同期して、図示しないCRTに表示するアドレスを発生し、ディスプレイバッファ21から表示データを読み出す要求をメモリI/F回路13に出力する。この要求に応じて、メモリI/F回路13は、ディスプレイバッファ21から一定の固まりで表示データを読み出す。CRTコントローラ回路14は、ディスプレイバッファ21から読み出した表示データを記憶するFIFO(First In First Out)回路を内蔵し、一定の時間間隔で、RAMDAC回路15に、RGBのインデックス値を出力する。

【0060】

RAMDAC回路15

RAMDAC回路15は、各インデックス値に対応するR, G, Bデータを記憶しており、CRTコントローラ回路14から入力したRGBのインデックス値に対応するデジタル形式のR, G, Bデータを、D/Aコンバータに転送し、アナログ形式のR, G, Bデータを生成する。RAMDAC回路15は、この生成されたR, G, BデータをCRTに出力する。

【0061】

以下、3次元コンピュータグラフィックシステム1の動作について説明する。

図1に示す3次元コンピュータグラフィックシステム1では、ポリゴンレンダリングデータS4が、メインバス6を介してメインプロセッサ4からDDAセットアップ回路10に出力され、DDAセットアップ回路10において、三角形の辺と水平方向の差分を示す変分データS10が生成される。

そして、DDAセットアップ回路10からトライアングルDDA回路11に変分データS10が出力される。

【0062】

次に、トライアングルDDA回路11において、変分データS10に基づいて、三角形内部の各画素における線形補間された(z, R, G, B, s, t, q, F)が生成される。

そして、トライアングルDDA回路11からテクスチャエンジン回路12に、各画素の(x, y)データと、当該(x, y)座標における(z, R, G, B, s, t, q, F)データとが、DDAデータS11として出力される。

【0063】

次に、図2に示すテクスチャエンジン回路12の縮小率演算回路304において、DDAデータS11に含まれる8画素分の(s, t, q)データS11a₁ ~ S11a₈を用い

10

20

30

40

50

て、テクスチャデータの縮小率が算出され、この縮小率 $l o d$ がテクスチャデータ読み出し回路 305 に出力される。

【0064】

次に、テクスチャデータ読み出し回路 305 において、図 5 に示すフローに基づいて、上記式 (3) に基づいて生成された 1 次元の物理アドレス A を用いて、テクスチャバッファ 20 (SRAM17) からテクスチャデータ $S 17_1 \sim S 17_8$ が読み出され、この読み出されたテクスチャデータ $S 17_1 \sim S 17_8$ が、テクスチャブレンド回路 306 に出力される。

【0065】

このとき、図 7 に示す読み出しコントローラ 262 からの制御によって、配線群 280 を介して、テクスチャデータ $S 17_1 \sim S 17_8$ を含む 16 画素分のテクスチャデータが、SRAM17 を構成するバンク $220_1, 220_2, 221_1, 221_2, 222_1, 222_2, 223_1, 223_2, 224_1, 224_2, 225_1, 225_2, 226_1, 226_2, 227_1, 227_2$ から読み出される。

【0066】

次に、テクスチャブレンド回路 306 において、(R, G, B) データ $S 11b_1 \sim S 11b_8$ と、データ $S 17_1 \sim S 17_8$ に含まれる (R, G, B) データとが、データ $S 17_1 \sim S 17_8$ に含まれる t で示される混合値で混合され、(R, G, B) データ $S 306_1 \sim S 306_8$ が生成される。

そして、DDA データに含まれるデータ $S 11d_1 \sim S 11d_8$ と、(R, G, B) データ $S 306_1 \sim S 306_8$ とが、(R, G, B) データ $S 12a_1 \sim S 12a_8$ 、すなわち、画素データ $S 12a$ として、メモリ I/F 回路 13 に出力される。

【0067】

そして、メモリ I/F 回路 13 において、テクスチャエンジン回路 12 から入力した画素データ $S 12a$ に対応する z データと、 z バッファ 22 に記憶されている z データとの比較が行なわれ、入力した画素データ $S 12a$ によって描画される画像が、前回、ディスプレイバッファ 21 に書き込まれた画像より、手前 (視点側) に位置するか否かが判断され、手前に位置する場合には、画像データ $S 12a$ に対応する z データで z バッファ 22 に記憶された z データが更新される。

【0068】

次に、メモリ I/F 回路 13 において、必要に応じて、画像データ $S 12a$ に含まれる (R, G, B) データと、既にディスプレイバッファ 21 に記憶されている (R, G, B) データとが、画素データ $S 12a$ に対応するデータが示す混合値で混合され、混合後の (R, G, B) データがディスプレイバッファ 21 に書き込まれる。

【0069】

このとき、図 7 に示すメモリコントローラ 240, 341, 242, 243 からの制御によって、配線群 270, 271, 272, 273 を介して、16 画素分の (R, G, B) データが、図 1 に示すディスプレイバッファ 21 を構成するバンク $210_1, 210_2, 211_1, 211_2, 212_1, 212_2, 213_1, 213_2, 214_1, 214_2, 215_1, 215_2, 216_1, 216_2, 217_1, 217_2$ に書き込まれる

【0070】

以上説明したように、3 次元コンピュータグラフィックシステム 1 によれば、テクスチャバッファ 20 の 2 次元アドレス空間を示す 2 次元のテクスチャアドレス (U, V) の U および V を構成する各ビットを、上記式 (3) に基づいて組み合わせることで、1 次元の物理アドレス A を生成できる。ここで、物理アドレス A の生成は、簡単なビット操作で実現できることから、当該物理アドレス A の生成を、小規模な回路構成で高速に実現できる。また、1 次元の物理アドレスを用いてテクスチャバッファ 20 の連続した記憶領域にテクスチャを記憶できることから、図 10 に示すような空き領域が生じることはなく、テクスチャバッファ 20 の記憶領域を効率的に使用できる。その結果、テクスチャバッファ 20 の記憶容量を小さくでき、装置の小規模化および低価格化を図れる。

10

20

30

40

50

【 0 0 7 1 】

本発明は上述した実施形態には限定されない。

例えば、上述した実施形態では、 U_i および V_i が共に 3 ビットである場合を例示したが、本発明は、 U_i および V_i が共に、 $n = 2$ 、あるいは、 $n (n \geq 4)$ ビットの場合にも適用可能である。

例えば、 U_i および V_i が、下記式 (4)、(5) に示すように、共に 4 ビットの場合には、下記式 (5) によって 1 次元の物理アドレス A が図 8 に示すように、生成される。

【 0 0 7 2 】

【数 4】

$$U_i = \{ U_i [3], U_i [2], U_i [1], U_i [0] \} \dots (4)$$

10

【 0 0 7 3 】

【数 5】

$$V_i = \{ V_i [3], V_i [2], V_i [1], V_i [0] \} \dots (5)$$

【 0 0 7 4 】

【数 6】

$$\text{物理アドレス } A = (V_i [3], U_i [3], V_i [2], U_i [2], V_i [1], U_i [1], V_i [0], U_i [0]) \dots (6)$$

【 0 0 7 5 】

また、 U_i および V_i のビット数が一致しなくてもよい。具体的には、 U_i が n ビットのときに、 V_i が $n - 1$ ビットであってもよい。

20

例えば、 U_i および V_i が、下記式 (7)、(8) に示すように、それぞれ 3 ビットおよび 2 ビットの場合には、下記式 (9) によって 1 次元の物理アドレス A が図 9 に示すように生成される。

【 0 0 7 6 】

【数 7】

$$U_i = \{ U_i [2], U_i [1], U_i [0] \} \dots (7)$$

【 0 0 7 7 】

【数 8】

$$V_i = \{ V_i [1], V_i [0] \} \dots (8)$$

【 0 0 7 8 】

【数 9】

$$\text{物理アドレス } A = (U_i [3], V_i [2], U_i [2], V_i [1], U_i [1], V_i [0], U_i [0]) \dots (9)$$

30

【 0 0 7 9 】

また、上述した実施形態では、同時に処理が実行される画素数を 8 としたが、この数は任意であり、例えば、4 であってもよい。但し、同時に処理が実行される画素数は、2 のべき乗であることが望ましい。

【 0 0 8 0 】

また、上述した図 1 に示す 3 次元コンピュータグラフィックシステム 1 では、SRAM 17 を用いる構成を例示したが、SRAM 17 を設けない構成にしてもよい。

40

また、図 1 に示すテクスチャバッファ 20 およびテクスチャ CLUT バッファ 23 を、DRAM 16 の外部に設けてもよい。

【 0 0 8 1 】

さらに、図 1 に示す 3 次元コンピュータグラフィックシステム 1 では、ポリゴンレンダリングデータを生成するジオメトリ処理を、メインプロセッサ 4 で行なう場合を例示したが、レンダリング回路 5 で行なう構成にしてもよい。

【 0 0 8 2 】

【発明の効果】

以上説明したように、本発明の記憶回路制御装置によれば、小規模な回路構成で、2 次元画像データを記憶回路に効率的に記憶でき、記憶回路の記憶領域を有効利用できる。

50

また、本発明のグラフィック演算装置によれば、小規模な回路構成で、テクスチャデータを記憶回路に効率的に記憶でき、記憶回路の記憶領域を有効利用できる。

また、本発明の記憶回路制御方法によれば、2次元画像データを記憶回路に効率的に記憶でき、記憶回路の記憶領域を有効利用できる。

また、本発明のグラフィック演算方法によれば、テクスチャデータを記憶回路に効率的に記憶でき、記憶回路の記憶領域を有効利用できる。

また、本発明の記憶回路制御装置およびその方法と、グラフィック演算装置およびその方法とによれば、記憶回路にアクセスする際のアドレス変換を高速に行なうことができる。

【図面の簡単な説明】

【図1】図1は、本発明の実施形態の3次元コンピュータグラフィックシステムのシステム構成図である。 10

【図2】図2は、図1に示すテクスチャエンジン回路の内部構成図である。

【図3】図3は、図1に示すテクスチャバッファに記憶され、MIPMAPフィルタリング処理された複数の縮小率のテクスチャデータを説明するための図である。

【図4】図4は、テクスチャデータ読み出し回路の構成図である。

【図5】図5は、図2に示すテクスチャデータ読み出し回路における処理のフローチャートである。

【図6】図6は、テクスチャエンジン回路において、2次元のテクスチャアドレス(U_i, V_i)から1次元の物理アドレスAを生成する方法を説明するための図である。

【図7】図7は、図1に示すDRAM、SRAM、および、メモリI/F回路のDRAMおよびSRAMへのアクセス機能を持つブロックの構成図である。 20

【図8】図8は、2次元のテクスチャアドレス(U, V)が共に、4ビットの場合における1次元の物理アドレスAの生成方法を説明するための図である。

【図9】図9は、2次元のテクスチャアドレス(U, V)のUが3ビットであり、Vが2ビットのである場合における1次元の物理アドレスAの生成方法を説明するための図である。

【図10】図10は、2次元のテクスチャアドレス(U, V)を直接用いて、複数の種類のテクスチャデータをテクスチャバッファに記憶する場合の問題点を説明するための図である。

【図11】図11は、1次元の物理アドレスを用いて、複数の種類のテクスチャデータをテクスチャバッファに記憶したときの記憶状態を説明するための図である。 30

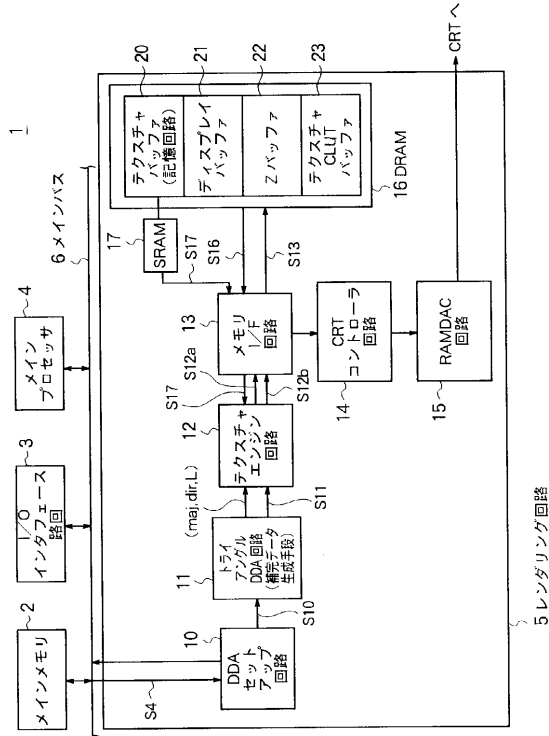
【図12】図12は、従来の3次元コンピュータグラフィックシステムの部分構成図である。

【図13】図13は、従来の高速処理が可能な3次元コンピュータグラフィックシステムの部分構成図である。

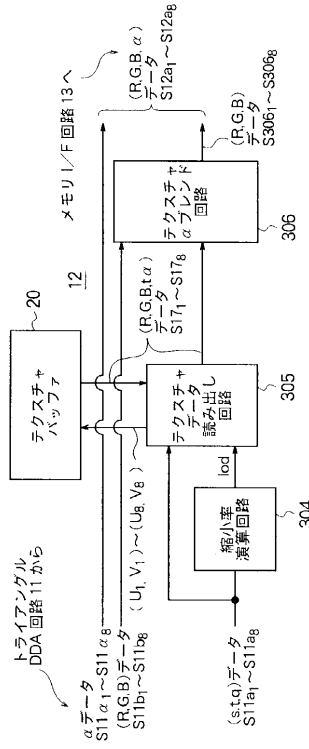
【符号の説明】

1...3次元コンピュータグラフィックシステム、2...メインメモリ、3...I/Oインタフェース回路、4...メインプロセッサ、5...レンダリング回路、10...DDAセットアップ回路、11...トライアングルDDA回路、12...テクスチャエンジン回路、13...メモリI/F回路、14...CRTコントローラ回路、15...RAMDAC回路、16...DRAM、17...SRAM、20...テクスチャバッファ、21...ディスプレイバッファ、22...Zバッファ、23...テクスチャCLUTバッファ、304...縮小率演算回路、305...テクスチャデータ読み出し回路、306...テクスチャブレンド回路、200, 201, 202, 203...メモリモジュール、210, 211, 212, 213, 214, 215, 216, 217...メモリ、240, 241, 242, 243...メモリコントローラ、250, 251, 252, 253...アドレスコンバータ、260...ディストリビュータ、262...読み出しコントローラ、270, 271, 272, 273, 280...配線群 40

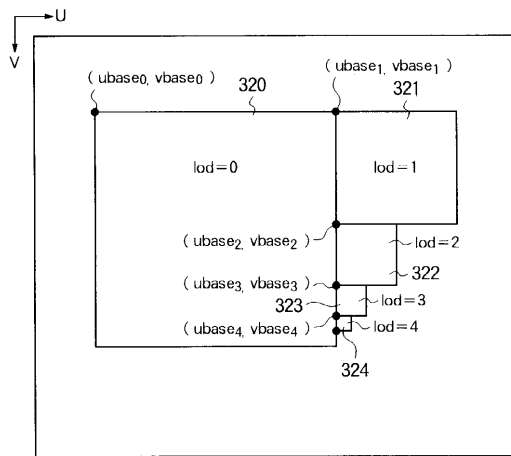
【図1】



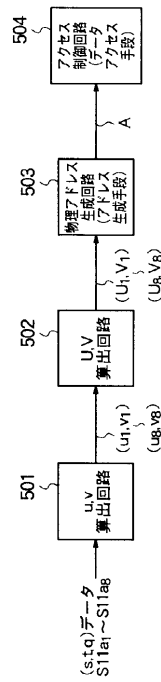
【図2】



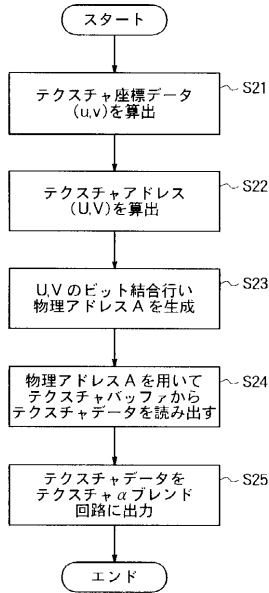
【図3】



【図4】



【図5】

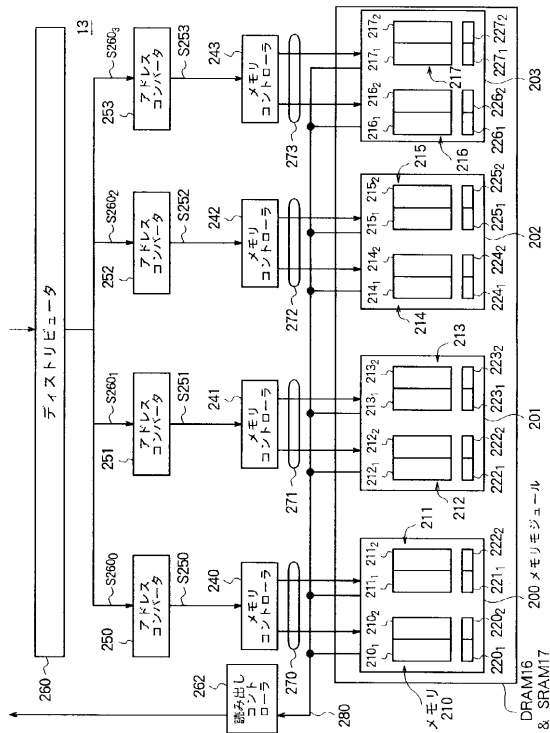


【図6】

	U →								
	B ↓	0	1	2	3	4	5	6	7
0		0	1	4	5	16	17	20	21
1		2	3	6	7	18	19	22	23
2		8	9	12	13	24	25	28	29
3		10	11	14	15	26	27	30	31
4		32	33	36	37	48	49	52	53
5		34	35	38	39	50	51	54	55
6		40	41	44	45	56	57	60	61
7		42	43	46	47	58	59	62	63

A

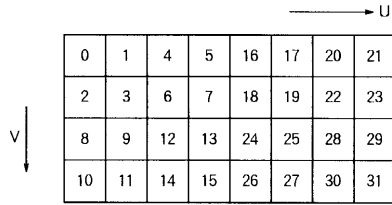
【図7】



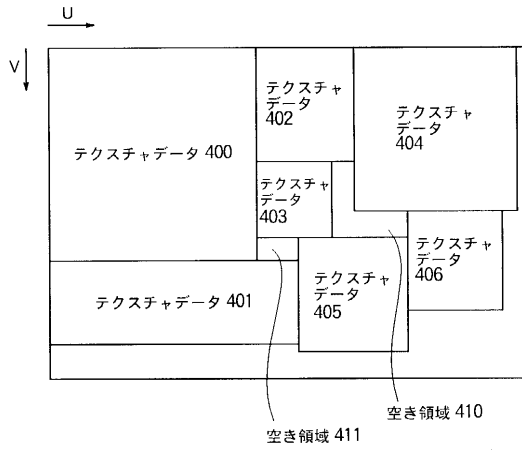
【図8】

	U →															
	0	1	4	5	16	17	20	21	64	65	68	69	80	81	84	85
2	2	3	6	7	18	19	22	23	66	67	70	71	82	83	86	87
8	8	9	12	13	24	25	28	29	72	73	76	77	88	89	92	93
10	10	11	14	15	26	27	30	31	74	75	78	79	90	91	94	95
32	32	33	36	37	48	49	52	53	96	97	100	101	112	113	116	117
34	34	35	38	39	50	51	54	55	98	99	102	103	114	115	118	119
40	40	41	44	45	56	57	60	61	104	105	108	109	120	121	124	125
42	42	43	46	47	58	59	62	63	106	107	110	111	122	123	126	127
128	128	129	132	133	144	145	148	149	192	193	196	197	208	209	212	213
130	130	131	134	135	146	147	150	151	194	195	198	199	210	211	214	215
136	136	137	140	141	152	153	156	157	200	201	204	205	216	217	220	221
138	138	139	142	143	154	155	158	159	202	203	206	207	218	219	222	223
160	160	161	164	165	176	177	180	181	224	225	228	229	240	241	244	245
162	162	163	166	167	178	179	182	183	226	227	230	231	242	243	246	247
168	168	169	172	173	184	185	188	189	232	233	236	237	248	249	252	253
170	170	171	174	175	186	187	190	191	234	235	238	239	250	251	254	255

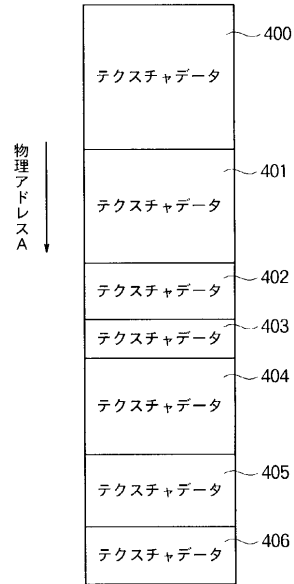
【図9】



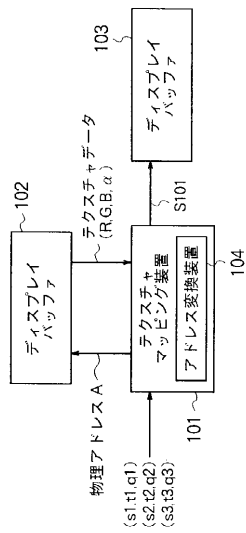
【図10】



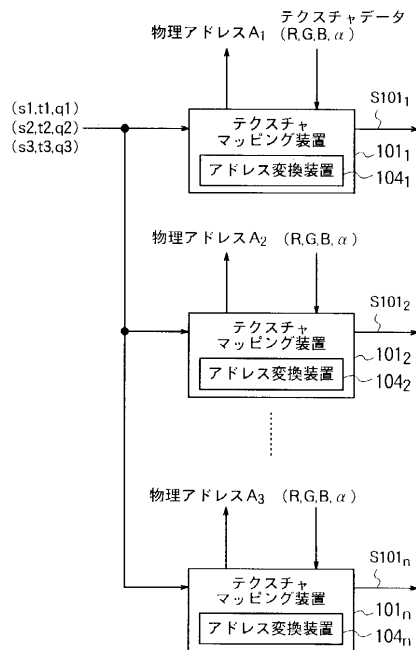
【図11】



【図12】



【図13】



フロントページの続き

(58)調査した分野(Int.Cl. , D B名)

G06T 15/00 - 17/50

CSDB(日本国特許庁)