

JP8101791

Publication Title:

VARIABLE-LENGTH BIT DATA PROCESSING CIRCUIT AND METHOD

Abstract:

A variable bit-length code processing circuit includes first, second and third registers (12, 20, 22) each of which is of 1 word, and memory data is loaded to the first register (12), and a variable bit-length code is withdrawn from the third register (22). The second register (20) and third register (22) are coupled to a barrel shifter (16) which barrel-shifts data of 2 words according to a barrel shift amount which is applied by a subtracter (30) on the basis of the number of the valid bits and the number of the remaining bits.

Data supplied from the esp@cenet database - <http://ep.espacenet.com>

【特許請求の範囲】

【請求項 1】ビット並列に与えられるデータを保持する 1 ワードの第 1 レジスタ、
前記第 1 レジスタからのデータをビット並列に受ける 1 ワードの第 2 レジスタ、
可変長ビットデータが取り出される 1 ワードの第 3 レジスタ、
有効ビット数と前記第 2 レジスタに保持されたデータの残ビット数とを比較して前記有効ビット数が前記残ビット数より大きいとき第 1 信号を出力する比較手段、
前記比較手段から前記第 1 信号が出力されたとき、前記第 2 レジスタおよび前記第 3 レジスタから前記残ビット数だけバレルシフトしたデータを前記第 3 レジスタへロードする第 1 バレルシフト手段、
前記比較手段から前記第 1 信号が出力されたとき、前記第 1 レジスタのデータを前記第 2 レジスタにビット並列でロードするロード手段、および前記ロード手段によって前記第 2 レジスタに前記第 1 レジスタのデータをロードした後、前記有効ビット数と前記残ビット数との差のビット数だけバレルシフトした前記第 2 レジスタおよび前記第 3 レジスタのデータを前記第 2 レジスタおよび前記第 3 レジスタにロードする第 2 バレルシフト手段を備える、可変長ビットデータ処理回路。

【請求項 2】前記比較手段は前記有効ビット数と前記残ビット数とが等しいとき第 2 の信号を出力し、
前記比較手段から前記第 2 信号が出力されたとき、前記残ビット数だけバレルシフトした前記第 2 レジスタおよび前記第 3 レジスタのデータを前記第 3 レジスタにロードする第 3 バレルシフト手段、および前記比較手段から前記第 2 信号が出力されたとき、前記第 1 レジスタのデータを前記第 2 レジスタにビット並列でロードするロード手段を備える、請求項 1 記載の可変長ビットデータ処理回路。

【請求項 3】前記比較手段は前記残ビット数が前記有効ビット数より大きいとき第 3 信号を出力し、
前記比較手段から前記第 3 信号が出力されたとき、前記第 2 レジスタおよび前記第 3 レジスタのデータを前記有効ビット数だけバレルシフトして前記第 2 レジスタおよび前記第 3 レジスタにロードする第 4 バレルシフト手段を備える、請求項 1 または 2 記載の可変長ビットデータ処理回路。

【請求項 4】ビット並列に与えられるデータを保持する 1 ワードの第 1 レジスタ、前記第 1 レジスタからのデータをビット並列に受ける 1 ワードの第 2 レジスタ、および可変長ビットデータが取り出される 1 ワードの第 3 レジスタを備える可変長ビットデータ処理回路における処理方法であって、
有効ビット数が前記第 2 レジスタに保持されたデータの残ビット数より大きいとき、(a) 前記第 2 レジスタおよび前記第 3 レジスタから前記残ビット数だけバレルシフ

トしたデータを前記第 3 レジスタへロードし、(b) 前記第 1 レジスタのデータを前記第 2 レジスタにビット並列でロードし、そして(c) 前記第 2 レジスタおよび前記第 3 レジスタのデータを前記有効ビット数と前記残ビット数との差のビット数だけバレルシフトして前記第 2 レジスタおよび前記第 3 レジスタにロードする、処理方法。

【請求項 5】前記有効ビット数と前記残ビット数とが等しいとき、(d) 前記残ビット数だけバレルシフトした前記第 2 レジスタおよび前記第 3 レジスタのデータを前記第 2 レジスタおよび前記第 3 レジスタにロードし、そして(e) 前記第 1 レジスタのデータを前記第 2 レジスタにビット並列でロードする、請求項 4 記載の処理方法。

【請求項 6】前記残ビット数が前記有効ビット数より大きいとき、(f) 前記第 2 レジスタおよび前記第 3 レジスタのデータを前記有効ビット数だけバレルシフトして前記第 2 レジスタおよび前記第 3 レジスタにロードする、請求項 4 または 5 記載の処理方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は可変長ビットデータ処理回路および方法に関し、特にたとえば、画像や音声の圧縮データのように、メモリに効率よく（余白なしに）格納されている可変長ビットデータを処理する、処理回路および方法に関する。

【0002】

【従来の技術】この種の可変長ビットデータ処理回路の一例が、平成 4 年（1992）9 月 2 日付で出願公開された特開平 4-245778 号公報に開示されている。この従来技術は、入力符号データを保持する 1 ワードのシフトレジスタ(11)と、そのシフトレジスタ(11)からビット並列に転送されるデータを保持する 3 ワードのシフトレジスタ(12)とを含み、シフト／転送制御部(30)がシフトレジスタ(12)内のデータ余白部分を充足させる分だけシフトレジスタ(11)内のデータを切り出して転送するものである。

【0003】

【発明が解決しようとする課題】従来技術では、不要になった前回のデータをシフトレジスタ(12)で直列シフトして排出するようにしているため、合計 4 ワード分のシフトレジスタと、それらのシフト動作を制御するためのシフト／転送制御とが必要となる。シフト／転送制御の詳細については開示されていないが、シフト回数を計数するカウンタ、シフトクロックの制御回路、シフトレジスタ 11 からシフトレジスタ 12 への任意のビット数のデータを並列転送する転送制御回路、およびシーケンサ等が必要となり、回路規模が大きくなってしまふ。さらに、次のデータとビットストリームでつなぐために余白ビットをシフトレジスタ(11)によって押し出した分だけシフトレジスタ(12)によってシフトしかつビット単位にシフトレジスタ(11)に書き込むようにしているため、従

来技術では、処理ステップが多くなり、データを取り出す時間が長くなってしまいう問題点があった。

【0004】それゆえに、この発明の主たる目的は、簡単な回路でかつより高速に、可変長ビットデータを取り出すことができる、データ処理回路および方法を提供することである。

【0005】

【課題を解決するための手段】この発明は、ビット並列に与えられるデータを保持する1ワードの第1レジスタ、第1レジスタからのデータをビット並列に受ける1ワードの第2レジスタ、可変長ビットデータが取り出される1ワードの第3レジスタ、有効ビット数と第2レジスタに保持されたデータの残ビット数とを比較して有効ビット数が残ビット数より大きいとき第1信号を出力する比較手段、比較手段から第1信号が出力されたとき、第2レジスタおよび第3レジスタから残ビット数だけバレルシフトしたデータを第3レジスタへロードする第1バレルシフト手段、比較手段から第1信号が出力されたとき、第1レジスタのデータを第2レジスタにビット並列でロードするロード手段、およびロード手段によって第2レジスタに第1レジスタのデータをロードした後、有効ビット数と残ビット数との差のビット数だけバレルシフトした第2レジスタおよび第3レジスタのデータを第2レジスタおよび第3レジスタにロードする第2バレルシフト手段を備える、可変長ビットデータ処理回路である。

【0006】この発明は、また、ビット並列に与えられるデータを保持する1ワードの第1レジスタ、第1レジスタからのデータをビット並列に受ける1ワードの第2レジスタ、および可変長ビットデータが取り出される1ワードの第3レジスタを備える可変長ビットデータ処理回路における処理方法であって、有効ビット数が第2レジスタに保持されたデータの残ビット数より大きいとき、(a) 第2レジスタおよび第3レジスタから残ビット数だけバレルシフトしたデータを第3レジスタへロードし、(b) 第1レジスタのデータを第2レジスタにビット並列でロードし、そして(c) 第2レジスタおよび第3レジスタのデータを有効ビット数と残ビット数との差のビット数だけバレルシフトして第2レジスタおよび第3レジスタにロードする、処理方法である。

【0007】

【作用】たとえばメモリから読み出された1ワードのデータが第1レジスタにビット並列にロードされる。この第1レジスタのデータが第2レジスタにビット並列にロードされ、第2レジスタのデータが第1バレルシフト手段によってたとえば16ビットシフトされて第3レジスタにロードされる。第3レジスタから、可変長ビットデータが取り出される。たとえば減算器からなる比較手段において、たとえばCPUによって設定された有効ビット数を第2レジスタに保持されたデータの残ビット数か

ら減算する。そして、その減算結果が負になったとき、減算器からボロー信号(第1信号)が出力される。このボロー信号が出力されたとき、バレルシフタが動作し、第2レジスタおよび第3レジスタのデータを残ビット数だけバレルシフトしてそれを第3レジスタにロードする。このとき、第1バレルシフト手段のバレルシフトによってはみ出したデータは失われ、使用されない。それとともに、第1レジスタのデータが第2レジスタにロードされ、その第2レジスタのデータが第3レジスタのデータとともに(有効ビット数-残ビット数)だけ第2バレル手段によってバレルシフトされて第2レジスタおよび第3レジスタにロードされる。第2バレルシフト手段のバレルシフトによって空きが生じた第2レジスタの空白部には「0」が書き込まれる。

【0008】

【発明の効果】この発明によれば、合計3ワードのラッチと、バレルシフト手段を構成するための複数のセレクトアや減算器があればよく、従来技術に比べて、回路構成が簡単になるとともに、多くても2ステップのバレルシフト動作で有効データがセットできるので、処理時間が短縮され、データ処理が一層高速化される。

【0009】この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

【0010】

【実施例】図1に示す可変長ビットデータ処理回路10は、メモリ(図示せず)から出力される可変長ビットデータを処理するための回路であって、そのメモリからの1ワードのメモリデータをビット並列に受ける1ワード(16ビット)の第1レジスタ12を含む。この第1レジスタ12のデータは、ビット並列にセレクトア14に与えられる。セレクトア14は2入力(各16ビット)のいずれかを選択して出力するものであって、一方入力として上述の第1レジスタ12からの16ビットが、他方入力としてバレルシフタ16の上位16ビットが与えられる。セレクトア14は、タイミング発生器18からの「0」または「1」のセレクト信号に応じて、第1レジスタ12からのデータまたはバレルシフタ16からのデータを選択的に、第2レジスタ20にビット並列にロードする。したがって、このセレクトア14は、第1ないし第4バレルシフト手段の一部を構成するとともに、第1レジスタ12のデータを第2レジスタ20にロードするロード手段として機能する。第2レジスタ20は1ワード(16ビット)のレジスタであり、この第2レジスタ20のデータはビット並列にバレルシフタ16に与えられる。第3レジスタ22もまた1ワード(16ビット)のレジスタである。なお、便宜上、第2レジスタ20および第3レジスタ22は個別のものとして説明したが、実際には、2ワード(32ビット)の1つのレジスタないしラッチとして構成される。第2レジスタ20のデー

5

タがバレルシフタ16の上位16ビットに与えられ、第3レジスタ22のデータがバレルシフタ16の下位16ビットに与えられる。バレルシフタ16は、第2レジスタ20および第3レジスタ22のそれぞれの16ビット並列データを1つの32ビットデータとして右方向にバレルシフトする。バレルシフトしたデータの上位16ビットは上述のようにセクタ14に与えられるが、バレルシフトしたデータの下位16ビットは第3レジスタ22にビット並列で与えられる。したがって、このバレルシフタ16が第1ないし第4バレルシフト手段の一部を構成することは明らかである。その結果、第3レジスタ22のデータが、CPU（図示せず）によって有効データとして3状態ゲート24から取り出される。

【0011】ここで、バレルシフトについて説明する。通常のシフトレジスタを用いたデータシフトでは、1回与えられるクロックにตอบสนองしてデータが1ビット分だけ右または左にシフトされる。たとえば、データを5ビットシフトする場合は、5つのクロックにตอบสนองしてデータが右または左に5ビットシフトされる。これに対して、ここでいうバレルシフトを用いた場合、1つのクロックにตอบสนองして、データが1度に複数ビット右または左にシフトされる。1回のクロックでシフトするビット数は、減算器30（後述）から与えられるシフト量により決定される。

【0012】このようなバレルシフトを行うために、この実施例のバレルシフタ16は、図2に示すように、32個のセクタSBL0-SBL31を用いて構成することができる。バレルシフタ16には、第3レジスタ22からの16ビットの並列データb16-b31と第2レジスタ20からの16ビットの並列データb0-b15の計32ビットのデータが与えられる。この32ビットのデータの各ビットを、下位から上位にかけてそれぞれb0-b31とする。

【0013】一方、先に述べたように、バレルシフタ16は、最大で16ビットのバレルシフトを行うことができる。このバレルシフトは、セクタSBL0-SBL31によって実現される。以下に、各セクタSBL0-SBL31の動作について説明する。バレルシフタ16の最下位ビットの出力を決定するセクタSBL0は、ビットb1-b16の中から1ビットを選択して出力する16-1セクタ（16ビットの中から任意の1ビットを選択して出力するセクタ）であり、選択するビットは減算器30（図1）から与えられるシフト量にตอบสนองして決定される。同様に、セクタSBL1, SBL2, SBL3, …SBL15は、それぞれ、ビットb2-b17, b3-b18, b4-b19, …b16-b31の中から、減算器30から与えられるシフト量にตอบสนองして、1ビットを選択して出力する16-1セクタである。

【0014】一方、セクタSBL16-SBL31の各々は、セクタSBL0-SBL15に用いたような16-1セクタによって構成することができない。なぜならば、セクタSBL16は、ビットb17-b32の中から1ビットを選択して出

6

力することができないからである。その理由は、バレルシフタ16には、32ビットb0-b31のデータしか与えられておらず、バレルシフタ16に与えられていないビットb32を選択することはできないからである。そこで、セクタSBL16は、減算器30から与えられるシフト量によって16ビットのバレルシフトを行うよう指示された場合は、ビットb32に代えて「0」を出力する。すなわち、セクタSBL16は、ビットb17-b31または「0」の中から1ビットを選択して出力する16-1セクタである。また、セクタSBL17は、バレルシフタ16にビットb32およびb33が与えられていないため、ビットb18-b33の中から1ビットを選択して出力することができず、したがって、減算器30によって15ビット以上のバレルシフトを行うよう指示された場合、すなわち、ビットb32あるいはb33を選択するよう指示された場合は、「0」を出力する。換言すれば、セクタSBL17は、ビットb18-b31または「0」の中から1ビットを選択して出力する15-1セクタである。同様に、セクタSBL18, SBL19, SBL20, …SBL30は、それぞれ、ビットb19-b31または「0」、ビットb20-b31または「0」、ビットb21-b31または「0」、…ビットb31または「0」の中から1ビットを選択して出力する14-1セクタ、13-1セクタ、12-1セクタ、…2-1セクタである。そして、セクタSBL31は、シフト量の値によらず「0」しか出力しない1-1セクタであり、セクタSBL31の出力は「0」に固定されている。

【0015】このようにして、バレルシフタ16は、減算器30（図1）によって指示されたシフト量にに応じて、第2レジスタ20および第3レジスタ22から与えられる32ビットb0-b31または「0」を選択し、結果的に、所要のビット数だけバレルシフトし、そのバレルシフトの結果、はみ出したビットは失われ、空白となったビットには「0」が書き込まれる。

【0016】なお、第1レジスタ12には、メモリ（図示せず）からのメモリデータが、メモリデータライトクロック（後述）に応じてロードされ、セクタ14からのデータはタイミング発生器18から出力されるラッチクロックに応じて第2レジスタ20にロードされる。そして、3状態ゲート24はアドレスデコーダ（図示せず）から出力されるCPUリード信号によって制御される。すなわち、このCPUリード信号が「1」のとき3状態ゲート24からCPUリードデータが取り出される。上述のアドレスデコーダは、可変長ビットデータ（VBC: Variable Bit-length Code）の処理のスタートを示すVBCスタート信号を出力し、このVBCスタート信号が、上述のタイミング発生器18に与えられるとともに、イニシャルカウンタ26に与えられる。イニシャルカウンタ26はまた、メモリデータライトクロックを受け、それにに応じてインクリメントされる。タイミ

ング発生器18は、VBCスタート信号に応じてイネーブルされ、VBCリクエスト信号を発生する。このVBCリクエスト信号は、メモリデータライトクロックに同期して発生される。

【0017】CPU（図示せず）からのCPUライトデータが有効ビット数レジスタ28に与えられるとともに、タイミング発生器18に与えられる。有効ビット数データは、たとえば4ビットで与えられ、パレルシフト量を決定するために用いられる。有効ビット数データが「0000」のとき「16」の有効ビット数を表し、有効ビット数データが「0001」のとき「1」の有効ビット数を表し、有効ビット数データが「0010」のとき「2」の有効ビット数を表し、以下同様に、有効ビット数データが「1111」のとき「15」の有効ビット数を表す。また、アドレスデコーダ（図示せず）からの有効ビット数ライトクロックが有効ビット数レジスタ28に与えられるとともに、タイミング発生器18に与えられる。

【0018】有効ビット数レジスタ28にロードされた有効ビット数のデータは減算器30の一方入力に与えられる。減算器30の他方入力には、残ビット数レジスタ32から出力される残ビット数のデータが与えられる。すなわち、減算器30は「残ビット数-有効ビット数」の減算を実行するのであるから、第2レジスタ20の残ビット数を自身で知ることができる。したがって、減算器30から残ビット数データが出力され、タイミング発生器18から残ビット数ライトクロックが残ビット数レジスタ32に与えられることによって、残ビット数レジスタ32に残ビット数データが保持される。このようにして、減算器30は「残ビット数-有効ビット数」の演算を行い、その減算結果を新たな残ビット数データとして出力するとともに、後者が前者よりも大きいとき、「1」のボロー信号（第1信号）を出力し、前者と後者とが等しいとき「1」のゼロ信号（第2信号）を出力する。したがって、ボロー信号およびゼロ信号がともに「0」とき、減算器30からは第3信号が出力されたことになる。ボロー信号およびゼロ信号は、ともに、タイミング発生器18に与えられる。タイミング発生器18では、ボロー信号やゼロ信号に応じて、セレクト信号やラッチクロックを出力する。

【0019】なお、減算器30は、パレルシフト16に対して、パレルシフトすべきビット数すなわちシフト量を設定する。たとえば、第1信号が出力されたときにはシフト量として最初に残ビット数データが与えられ、ついで「有効ビット数-残ビット数」すなわち減算結果の絶対値が与えられ、第2信号が出力されたときにはシフト量として残ビット数データが与えられ、第3信号が出力されたときにはシフト量として有効ビット数データが与えられる。

【0020】アドレスカウンタ34には、アドレスデコ

ーダ36（図3）から与えられる初期アドレスライトクロックに応答して、メモリに記憶された可変長ビットデータの先頭アドレスデータがプリセットされる。アドレスカウンタ34にプリセットされたアドレスデータは、アービタ38から与えられるメモリデータライトクロックに応答してインクリメントされる。これにより、可変長ビットデータがメモリ内に複数ワードにわたって連続して記憶されている場合でも、アドレスカウンタ34に可変長ビットデータの先頭アドレスを1度プリセットするだけで、以後複数ワードにわたる可変長ビットデータを連続して読み出して第1レジスタ12にラッチすることができる。

【0021】アドレスセクタ40は、CPUから与えられるCPUアドレスバスのアドレスデータまたはアドレスカウンタ34から与えられるアドレスデータのいずれか一方を切り換えてメモリに与える、この切り換えは、アービタ38からのセレクト信号に基づいて行われる。アービタ38は、可変長ビットデータ処理回路10の動作および可変長ビットデータ処理回路10とCPUとによるメモリアクセスのタイミングを制御するものである。

【0022】アービタ38には、図1に示すようにシステムクロック、アドレスデコーダ36からのCPUメモリリクエスト信号、およびタイミング発生器18からのセレクト信号およびイニシャルフラグとが与えられる。アービタ38は、タイミング発生器18からのセレクト信号が、第1レジスタ12にラッチされているデータを第2レジスタ20に供給するようにセクタ14を切り換える信号であることを判断する。これに応答して、アービタ38は、アドレスカウンタ34をインクリメントさせるカウント信号を発生し、これに続いてメモリから与えられる可変長ビットデータを第1レジスタ12にラッチさせるメモリデータライトクロックを発生する。ただし、タイミング発生器18からのイニシャルフラグがイニシャル状態を示している場合は、カウント信号は発生されない。また、アービタ38は、メモリデータライトクロックを発生するのに先立って、アドレスセクタ40に対してアドレスカウンタ34からのアドレス信号をメモリに供給するよう指示するセレクト信号を発生する。

【0023】一方、アービタ38は、アドレスデコーダ36（図3）からのCPUメモリリクエスト信号を監視することにより、CPUがメモリへのアクセスを要求しているか否かを判断する。そして、CPUがメモリへのアクセスを要求し、かつアドレスセクタ40がアドレスカウンタ34からのアドレスデータをメモリに与えるよう切り換えられているときは、CPUに対してウェイト信号を与えて、CPUをウェイト状態にする。

【0024】ここで、図3を参照して、アドレスデコーダ36には、CPUアドレスバスとリード信号線とライ

ト信号線とが接続され、CPUからのアドレスデータとCPUからのリード信号とCPUからのライト信号とが与えられる。CPUは、所定の大きさのメモリ空間を有し、このメモリ空間内の所定のアドレスに上述のメモリ、可変長ビットデータ処理スタート、第3レジスタ22および有効ビット数レジスタ28等が割り当てられている。アドレスデコーダ36は、常に、CPUからのアドレスデータ、リード信号およびライト信号を監視し、CPUがメモリの割り当てられているアドレスをアクセスし、かつリード信号あるいはライト信号を出力しているときは、CPUメモリリクエスト信号を発生する。また、アドレスデコーダ36は、CPUが可変長ビットデータ処理をスタートさせるためのアドレスをアクセスしかつライト信号を出力しているときは、可変長ビットデータ処理回路10に可変長ビットデータ処理をスタートさせるためのVBCスタート信号を発生する。また、アドレスデコーダ36は、CPUがアドレスカウンタの割り当てられているアドレスをアクセスしかつライト信号を出力しているときは、初期アドレスライトクロックを発生する。さらに、アドレスデコーダ36は、CPUが第3レジスタの割り当てられているアドレスをアクセスしかつリード信号を出力しているときは、CPUリード信号を発生する。また、アドレスデコーダ36はCPUが有効ビット数レジスタ28の割り当てられているアドレスをアクセスしかつライト信号を出力しているときは、有効ビット数ライトクロック信号を発生する。

【0025】図4および図5を参照して、図1実施例の動作を説明する。なお、図1実施例は、実際には、ハードウェアで構成されているが、図4および図5のフロー図のように動作する。したがって、図4および図5フロー図に従って動作するマイクロコンピュータによって図1回路が代替されてもよい。図4のステップS1からステップS7までは、初期設定動作である。すなわち、CPUかつしたがってアドレスデコーダ36（図3）からのVBCスタート信号に応答してタイミング発生器18がイネーブルされ、ステップS1において、タイミング信号発生器18は、VBCリクエスト信号を出力する。なお、アドレスデコーダ36は、上述のようにCPUから所定のアドレスデータが出力されたとき、VBCスタート信号を出力する。このとき、CPUは、同時に、メモリアドレスカウンタ34（図1）に対してメモリリードアドレスをセットする。応じて、ステップS2において、メモリ（図示せず）からメモリデータが読み出され、メモリデータライトクロックに応じてそのメモリデータが、第1レジスタ12にビット並列にロードされる。そして、メモリデータライトクロックに応じて、アドレスカウンタ34がインクリメントされる。なお、メモリ（図示せず）には、可変ビット長データ、すなわち、有効ビット数が変化するデータが効率よく（余白なしに）アドレス順次にストアされている。タイミング発

生器18は、このメモリデータライトクロックをカウントして、次のステップS3において、再びVBCリクエスト信号を発生する。したがって、次のステップS4において、メモリ（図示せず）のインクリメントされたアドレスからのメモリデータがビット並列に第1レジスタ12にロードされるとともに、タイミング発生器18からのセレクト信号が「1」であるため、第1レジスタ12のデータがセクタ14を介して、ビット並列に第2レジスタ20にロードされる。タイミング発生器18は、メモリデータライトクロックをさらにカウントして、次のステップS5において、再びVBCリクエスト信号を発生する。したがって、次のステップS6において、メモリ（図示せず）からのメモリデータがビット並列に第1レジスタ12にロードされるとともに、タイミング発生器18からのセレクト信号が「1」であるため、第1レジスタ12のデータがセクタ14を介して、ビット並列に第2レジスタ20にロードされる。タイミング発生器18からのイニシャルフラグが減算器30に与えられ、減算器30から「16」、すなわち「0000」がバレルシフト16に与えられる。第2レジスタ20のデータはバレルシフト16を介して第3レジスタ22にロードされる。イニシャルカウンタ26は、ここまで3つのメモリデータライトクロックをカウントしているので、次のステップS7では、イニシャルエンド信号を出力し、タイミング発生器18に与える。したがって、タイミング発生器18は、初期設定動作を終える。

【0026】このようにして、図6に示すように、第1レジスタ12、第2レジスタ20および第3レジスタ22に、それぞれ、データがロードされる。第3レジスタ22には、最下位ビットから順に、16ビットのデータ $a_0, a_1, a_2, a_3, a_4, b_0, b_1, b_2, b_3, b_4, b_5, b_6, c_0, c_1, c_2, c_3$ がロードされ、第2レジスタ20には、最下位ビットから順に、16ビットのデータ $c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}, c_{13}, d_0, d_1, d_2, d_3, d_4, d_5$ がロードされ、第1レジスタ12には、最下位ビットから順に、16ビットのデータ $d_6, e_0, f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, g_0, g_1, g_2$ がロードされた状態を考える。アルファベット a, b, c, \dots がそれぞれ異なるデータを示し、データ a の有効ビット数は5ビット、データ b の有効ビット数は7ビット、データ c の有効ビット数は14ビット、データ d の有効ビット数は7ビット、データ e の有効ビット数は1ビット、データ f の有効ビット数は11ビット、そしてデータ g の有効ビット数は未定である。

【0027】次のステップS8において、CPU（図示せず）は第3レジスタ22から有効データに a_0-a_4 をリードする。ステップS9で、CPUライトデータとして、データ a の有効ビット数である「5」がCPUから出力される。応じて、アドレスデコーダ36（図3）から有効ビット数ライトクロックが出力されるので、有効

ビット数レジスタ28に、「5」のデータが書き込まれる。

【0028】次のステップS10において、減算器30が「残ビット数-有効ビット数」の演算を実行する。ここで、残ビット数とは、第2レジスタ20に保持されているデータのうち、バレルシフトの結果生じた「0」以外のデータ、たとえば、図6における c_4-c_{13} および d_0-d_6 の16ビットのデータあるいは図7における c_9-c_{13} および d_0-d_5 の11ビットのデータを意味する。図6の例では、第2レジスタ20の残ビット数は「16」であるため、ステップS10の減算結果は「 $16-5=11$ 」で正となり、ステップS10では「YES」と判断される。減算器30からは、このとき、バレルシフト16におけるシフト量として有効ビット数である「5」をバレルシフト16に指示する。したがって、次のステップS11において、有効ビット数「5」だけバレルシフトしたデータを第2レジスタ20および第3レジスタ22にロードする。その結果が、図7に示される。図7では、第1レジスタ12のデータはそのまま変化せず、第3レジスタ22に保持されていた5ビットのデータaが第3レジスタ22から押し出される。したがって、第3レジスタ22には、16ビットのデータ $b_0, b_1, b_2, b_3, b_4, b_5, b_6, c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$ が保持される。なお、減算器30からのボロー信号もゼロ信号もともに「0」であるので、タイミング発生器18からは「0」のセレクト信号が出力される。応じて、セレクト14はバレルシフト16からのデータを選択する。したがって、第2レジスタ20には11ビットのデータ $c_9, c_{10}, c_{11}, c_{12}, c_{13}, d_0, d_1, d_2, d_3, d_4, d_5$ と5ビットの「0」が保持される。「0」は、バレルシフト16によって第2レジスタ20のデータを5ビットバレルシフトした結果、第2レジスタ20に書き込まれる。

【0029】図7の状態において、第3レジスタ22のデータbの有効ビット数は「7」であり、第2レジスタ20の残ビット数は「11」である。したがって、ここで再び、ステップS8からステップS11が繰り返し実行される。その結果、図8に示すように、第3レジスタ22からは7ビットのデータbが排出され、データbはCPUリードデータとして取り込まれる。そして、第3レジスタ22には、16ビットのデータ $c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}, c_{13}, d_0, d_1$ が保持され、第2レジスタ20には、4ビットのデータ d_2, d_3, d_4, d_5 と12ビットの「0」が保持される。先に5ビットの「0」があり、かつ新たに7ビットのバレルシフトが行われるのであるから、この段階で第2レジスタ20の「0」は12ビットになる。

【0030】図8の状態において、第3レジスタ22のデータcの有効ビット数は「14」であり、第2レジスタ20の残ビット数は「4」である。したがって、減算器30の減算の結果は「 $4-14=-10$ 」となり、ボ

ロー信号が出力され、ステップS10では「NO」と判断される。そのため、プロセスは図5のステップS12に進み、このステップS12において、「残ビット数-有効ビット数」の減算結果が「0」であるかどうかを判断する。上述の例では、減算結果は「-10」であるため、このステップS12では「NO」と判断され、したがって、プロセスはステップS13に進む。

【0031】ボロー信号に応じて、タイミング発生器18は「0」のセレクト信号を出力する。また、減算器30からは、減算結果が負になったので、シフト量として、残ビット数である「4」をバレルシフト16に与える。したがって、ステップS13では、図9に示すように、まず、第2レジスタ20および第3レジスタ22のデータを4ビットバレルシフトし、それを第2レジスタ20および第3レジスタ22にロードする。このとき、バレルシフト16の下位16ビットのデータは、 $c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}, c_{13}, d_0, d_1, d_2, d_3, d_4, d_5$ となり、上位16ビットは全て「0」になる。ここで、次にタイミング発生器18からの「1」のセレクト信号に回答して、セレクト14が第1レジスタ12のデータを選択しているため、第2レジスタ20には第1レジスタ12のデータがビット並列でロードされ、第2レジスタ20は16ビットのデータ $d_6, e_0, f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, g_0, g_1, g_2$ が保持される。

【0032】次のステップS14では、減算器30から「有効ビット数-残ビット数」のシフト量がバレルシフト16に指示される。図8の状態では、第3レジスタ22の有効ビット数は「14」であり、第2レジスタ20の残ビット数は「4」である。したがって、このときのシフト量は「10」である。したがって、ステップS14において、第2レジスタ20および第3レジスタ22のデータをバレルシフト16によって10ビットバレルシフトしたデータを第2レジスタ20および第3レジスタ22にセレクト14を通してロードする。その結果、第3レジスタ22には、16ビットのデータ $d_0, d_1, d_2, d_3, d_4, d_5, d_6, e_0, f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7$ が保持され、第2レジスタ20には6ビットのデータ $f_8, f_9, f_{10}, g_0, g_1, g_2$ と10ビットの「0」とが保持される。

【0033】次のステップS15においてタイミング発生器18からVBCリクエスト信号が出力され、応じて、メモリ（図示せず）から次のメモリデータが読み出される。したがって、ステップS16では、図10に示すように、メモリデータが第1レジスタ12にビット並列にロードされ、第1レジスタ12には16ビットのデータ $g_3, g_4, g_5, g_6, g_7, h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}$ が保持される。

【0034】なお、タイミング発生器18は、メモリから第1レジスタ12にデータをロードする毎に、メモリアドレスカウンタ34をインクリメントするため、可変長ビットデータ列をメモリ内にアドレス順にストアして

おくだけで、それぞれの変長ビットデータは自動的に順次メモリから第1レジスタ12へロードされる。再びステップS12が実行されると、図9の状態では、第3レジスタ22のデータdの有効ビット数は「7」で残ビット数は「4」であり、その結果、ステップS13からステップS16が実行される。すなわち、ステップS13を経て、図10に示すように、第3レジスタ22には16ビットのデータ $d_6, e_0, f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, g_0, g_1, g_2, g_3$ が保持され、第2レジスタ20には16ビットのデータ $g_3, g_4, g_5, g_6, g_7, h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}$ が保持される。同時に、7ビットのデータdがCPUリードデータとして取り込まれる。その後のステップS15およびステップS16を経て、図10に示すように、第3レジスタ22には16ビットのデータ $e_0, f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7, g_0, g_1, g_2$ が保持され、第2レジスタ20には、15ビットのデータ $g_4, g_5, g_6, g_7, h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}$ と1ビットの「0」が保持され、第1レジスタ12にはメモリ（図示せず）からメモリデータ $h_{11}, h_{12}, h_{13}, h_{14}, h_{15}, i_0, i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, j_0$ がロードされる。

【0035】図10の状態では、第3レジスタ22のデータeの有効ビット数は「1」であり、残ビット数は「15」であるので、次に実行されるステップS10で「YES」と判断され、再び、ステップS11が実行される。その結果、図11に示すように、第3レジスタ22には16ビットのデータ $f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7, g_0, g_1, g_2, g_3, g_4$ が保持され、第2レジスタ20には、14ビットのデータ $g_5, g_6, g_7, h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}$ と2ビットの「0」が保持される。

【0036】図11の状態では、第3レジスタ22のデータfの有効ビット数は「11」であり、残ビット数は「14」であるので、次に実行されるステップS10で「YES」と判断され、再び、ステップS11が実行される。その結果、図12に示すように、第3レジスタ22には16ビットのデータ $g_0, g_1, g_2, g_3, g_4, g_5, g_6, g_7, h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7$ が保持され、第2レジスタ20には、3ビットのデータ h_8, h_9, h_{10} と13ビットの「0」が保持される。このとき、第1レジスタ12のデータは $h_{11}, h_{12}, h_{13}, h_{14}, h_{15}, i_0, i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, j_0$ で不変である。

【0037】図12の状態では、第3レジスタ22のデータgの有効ビット数は「8」であり、残ビット数は「3」であるため、次に実行されるステップS10およびステップS12とともに、「NO」と判断される。そのため、ステップS13からステップS16までが再び実行される。その結果が図13に示される。図13の状態において、第3レジスタ22のデータhの有効ビット数は「16」であり、第2レジスタ20の残ビット数は「11」である。したがって、減算器30の減算の結果

は「 $11 - 16 = -5$ 」となり、ボロー信号が出力され、ステップS10では「NO」と判断される。そのため、ステップS12において、「残ビット数-有効ビット数」の減算結果が「0」であるかどうかを判断する。上述の例では、減算結果は「-5」であるため、このステップS12では「NO」と判断され、したがって、プロセスはステップS13に進む。

【0038】ボロー信号に応じて、タイミング発生器18は「0」のセレクト信号を出力する。また、減算器30からは、減算結果が負になったので、シフト量として、残ビット数である「11」をバレルシフト16に与える。したがって、ステップS13では、図14に示すように、まず、第2レジスタ20および第3レジスタ22のデータを11ビットバレルシフトし、それを第2レジスタ20および第3レジスタ22にロードする。このとき、バレルシフト16の下位16ビットのデータは、 $h_{11}, h_{12}, h_{13}, h_{14}, h_{15}, i_0, i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, j_0$ となり、上位16ビットのデータは全て「0」となる。ここで、次にタイミング発生器18からの「1」のセレクト信号に回答して、セレクト14が第1レジスタ12のデータを選択するため、第2レジスタ20には第1レジスタ12のデータがビット並列でロードされ、第2レジスタ20には16ビットのデータ $j_1, j_2, j_3, j_4, j_5, j_6, j_7, j_8, k_0, k_1, k_2, k_3, k_4, k_5, k_6, k_7$ が保持される。

【0039】次のステップS14では、減算器30から「有効ビット数-残ビット数」のシフト量がバレルシフト16に指示される。図14の状態では、第3レジスタ22の有効ビット数は「16」であり、第2レジスタ20の残ビット数は「11」である。したがって、このときのシフト量は「5」である。したがって、ステップS14において、第2レジスタ20および第3レジスタ22のデータをバレルシフト16によって5ビットバレルシフトしたデータを第2レジスタ20および第3レジスタ22にセレクト14を通してロードする。その結果、第3レジスタ22には、16ビットのデータ $i_0, i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, j_0, j_1, j_2, j_3, j_4, j_5$ が保持され、第2レジスタ20には11ビットのデータ $j_6, j_7, j_8, k_0, k_1, k_2, k_3, k_4, k_5, k_6, k_7$ と5ビットの「0」とが保持される。

【0040】次のステップS15においてタイミング発生器18からVBCリクエスト信号が出力され、応じて、メモリ（図示せず）から次のメモリデータが読み出される。したがって、ステップS16では、図15に示すように、メモリデータが第1レジスタ12にビット並列にロードされ、第1レジスタ12には16ビットのデータ $l_0, l_1, l_2, l_3, l_4, m_0, m_1, n_0, n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5$ が保持される。

【0041】図14の状態において、第3レジスタ22のデータiの有効ビット数は「10」であり、第2レジスタ20の残ビット数は「11」である。したがって、

減算器30の減算の結果は「 $11-10=1$ 」となり、ステップS10では「YES」と判断される。減算器30からは、このとき、パレルシフト16におけるシフト量として有効ビット数である「10」をパレルシフト16に指示する。したがって、次のステップS11において、有効ビット数「10」だけパレルシフトしたデータを第2レジスタ20および第3レジスタ22にロードする。その結果が、図15に示される。図15では、第1レジスタ12のデータはそのまま変化せず、第3レジスタ22に保持されていた10ビットのデータiが第3レジスタ22から押し出される。したがって、第3レジスタ22には、16ビットのデータ $j_0, j_1, j_2, j_3, j_4, j_5, j_6, j_7, j_8, k_0, k_1, k_2, k_3, k_4, k_5, k_6$ が保持される。なお、減算器30からのボロー信号もゼロ信号もともに「0」であるので、タイミング発生器18からは「0」のセレクト信号が出力される。応じて、セレクト14はパレルシフト16からのデータを選択する。したがって、第2レジスタ20には1ビットのデータ k_7 と15ビットの「0」が保持される。「0」は、パレルシフト16によって第2レジスタ20のデータを10ビットパレルシフトした結果、第2レジスタ20に書き込まれる。

【0042】図15の状態において、第3レジスタ22のデータjの有効ビット数は「9」であり、第2レジスタ20の残ビット数は「1」である。したがって、減算器30の減算の結果は「 $1-9=-8$ 」となり、ボロー信号が出力され、ステップS10では「NO」と判断される。そのため、ステップS12において、「残ビット数-有効ビット数」の減算結果が「0」であるかどうかを判断する。上述の例では、減算結果は「-8」であるため、このステップS12では「NO」と判断され、したがって、プロセスはステップS13に進む。

【0043】ボロー信号に応じて、タイミング発生器18は「0」のセレクト信号を出力する。また、減算器30からは、減算結果が負になったので、シフト量として、残ビット数である「1」をパレルシフト16に与える。したがって、ステップS13では、図16に示すように、まず、第2レジスタ20および第3レジスタ22のデータを1ビットパレルシフトし、それを第2レジスタ20および第3レジスタ22にロードする。このとき、パレルシフト16の下位16ビットのデータは、 $j_1, j_2, j_3, j_4, j_5, j_6, j_7, j_8, k_0, k_1, k_2, k_3, k_4, k_5, k_6, k_7$ となり、上位16ビットは全て「0」になる。ここで、次にタイミング発生器18からの「1」のセレクト信号に応答して、セレクト14が第1レジスタ12のデータを選択するため、第2レジスタ20には第1レジスタ12のデータがビット並列でロードされ、第2レジスタ20は16ビットのデータ $l_0, l_1, l_2, l_3, l_4, m_0, m_1, n_0, n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5$ が保持される。

【0044】次のステップS14では、減算器30から「有効ビット数-残ビット数」のシフト量がパレルシフ

タ16に指示される。図15の状態では、第3レジスタ22の有効ビット数は「9」であり、第2レジスタ20の残ビット数は「1」である。したがって、このときのシフト量は「8」である。したがって、ステップS14において、第2レジスタ20および第3レジスタ22のデータをパレルシフト16によって8ビットパレルシフトしたデータを第2レジスタ20および第3レジスタ22にセレクト14を通してロードする。その結果、第3レジスタ22に保持されていた9ビットのデータjが第3レジスタ22から押し出される。そして、図16に示すように、第3レジスタ22には、16ビットのデータ $k_0, k_1, k_2, k_3, k_4, k_5, k_6, k_7, l_0, l_1, l_2, l_3, l_4, m_0, m_1, n_0$ が保持され、第2レジスタ20には8ビットのデータ $n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5$ と8ビットの「0」とが保持される。

【0045】次のステップS15においてタイミング発生器18からVBCリクエスト信号が出力され、応じて、メモリ（図示せず）から次のメモリデータが読み出される。したがって、ステップS16では、図17に示すように、メモリデータが第1レジスタ12にビット並列にロードされ、第1レジスタ12は16ビットのデータ $s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, t_0, t_1, u_0, u_1, u_2, u_3, u_4, u_5$ が保持される。

【0046】再びステップS12が実行されると、図16の状態では、第3レジスタ22のデータkの有効ビット数は「8」で残ビット数は「8」であり、その結果、図5のステップS17からステップS19が実行される。まず、ステップS17において、タイミング発生器18からの「1」のセレクト信号に応答して、セレクト14が第1レジスタ12のデータを選択しているため、第2レジスタ20には第1レジスタ12のデータがビット並列でロードされ、第2レジスタ20には16ビットのデータ $s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, t_0, t_1, u_0, u_1, u_2, u_3, u_4, u_5$ が保持される。それとともに、減算器30から有効ビット数に相当するシフト量がパレルシフト16に指示される。図16の状態では、第3レジスタ22の有効ビット数は「8」である。したがって、このときのシフト量は「8」である。したがって、ステップS17において、第2レジスタ20および第3レジスタ22のデータをパレルシフト16によって8ビットパレルシフトしたデータを第2レジスタ20および第3レジスタ22にセレクト14を通してロードする。その結果、第3レジスタ22に保持されていた8ビットのデータkが第3レジスタ22から押し出される。そして、図17に示すように、第3レジスタ22には、16ビットのデータ $l_0, l_1, l_2, l_3, l_4, m_0, m_1, n_0, n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5$ が保持され、第2レジスタ20には16ビットのデータ $s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, t_0, t_1, u_0, u_1, u_2, u_3, u_4, u_5$ が保持される。

【0047】次のステップS18においてタイミング発

生器 18 から VBC リクエスト信号が出力され、応じて、メモリ（図示せず）から次のメモリデータが読み出される。したがって、ステップ S 19 では、図 18 に示すように、メモリデータが第 1 レジスタ 12 にビット並列にロードされ、第 1 レジスタ 12 には 16 ビットのデータ $u_6, u_7, v_0, v_1, v_2, v_3, v_4, w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, x_0$ が保持される。

【0048】図 17 の状態において、第 3 レジスタ 22 のデータ 1 の有効ビット数は「5」であり、第 2 レジスタ 20 の残ビット数は「16」である。したがって、減算器 30 の減算の結果は「 $16 - 5 = 11$ 」となり、ステップ S 10 では「YES」と判断される。減算器 30 からは、このとき、バレルシフタ 16 におけるシフト量として有効ビット数である「5」をバレルシフタ 16 に指示する。したがって、次のステップ S 11 において、有効ビット数「5」だけバレルシフトしたデータを第 2 レジスタ 20 および第 3 レジスタ 22 にロードする。その結果が、図 18 の上部で示される。この状態では、第 1 レジスタ 12 のデータはそのまま変化せず、第 3 レジスタ 22 に保持されていた 5 ビットのデータ 1 が第 3 レジスタ 22 から押し出される。したがって、第 3 レジスタ 22 には、16 ビットのデータ $m_0, m_1, n_0, n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}$ が保持される。なお、減算器 30 からのボロー信号もゼロ信号もともに「0」であるので、タイミング発生器 18 からは「0」のセレクト信号が出力される。応じて、セクタ 14 はバレルシフタ 16 からのデータを選択する。したがって、第 2 レジスタ 20 には 11 ビットのデータ $s_{11}, s_{12}, s_{13}, t_0, t_1, u_0, u_1, u_2, u_3, u_4, u_5$ と 5 ビットの「0」が保持される。「0」は、バレルシフタ 16 によって第 2 レジスタ 20 のデータを 10 ビットバレルシフトした結果、第 2 レジスタ 20 に書き込まれる。

【0049】図 18 の上部の状態において、第 3 レジスタ 22 のデータ m の有効ビット数は「2」であり、第 2 レジスタ 20 の残ビット数は「11」である。したがって、減算器 30 の減算の結果は「 $11 - 2 = 9$ 」となり、ステップ S 10 では「YES」と判断される。減算器 30 からは、このとき、バレルシフタ 16 におけるシフト量として有効ビット数である「2」をバレルシフタ 16 に指示する。したがって、次のステップ S 11 において、有効ビット数「2」だけバレルシフトしたデータを第 2 レジスタ 20 および第 3 レジスタ 22 にロードする。その結果が、図 18 の下部に示される。図 18 の下部では、第 1 レジスタ 12 のデータはそのまま変化せず、第 3 レジスタ 22 に保持されていた 2 ビットのデータ m が第 3 レジスタ 22 から押し出される。したがって、第 3 レジスタ 22 には、16 ビットのデータ $n_0, n_1, n_2, s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}$ が保持される。なお、減算器 30 からのボロー信号もゼロ信号もともに「0」であるので、タイミング発生器 18

からは「0」のセレクト信号が出力される。応じて、セクタ 14 はバレルシフタ 16 からのデータを選択する。したがって、第 2 レジスタ 20 には 9 ビットのデータ $s_{13}, t_0, t_1, u_0, u_1, u_2, u_3, u_4, u_5$ と 7 ビットの「0」が保持される。「0」は、バレルシフタ 16 によって第 2 レジスタ 20 のデータを 10 ビットバレルシフトした結果、第 2 レジスタ 20 に書き込まれる。

【0050】このようにして、上述の実施例によれば、最大 2 回のバレルシフト動作を実行するだけで第 3 レジスタ 22 から有効データが取り出される。

【図面の簡単な説明】

【図 1】この発明の一実施例を示すブロック図である。

【図 2】図 1 実施例のバレルシフタを詳細に示すブロック図である。

【図 3】図 1 実施例のアドレスデコーダを示すブロック図である。

【図 4】図 1 実施例の動作を示すフロー図である。

【図 5】図 1 実施例の動作を示すフロー図である。

【図 6】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 7】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 8】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 9】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 10】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 11】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 12】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 13】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 14】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 15】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 16】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 17】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【図 18】図 1 実施例の動作を示す各レジスタの状態を示す図解図である。

【符号の説明】

10 …可変長ビットデータ処理回路

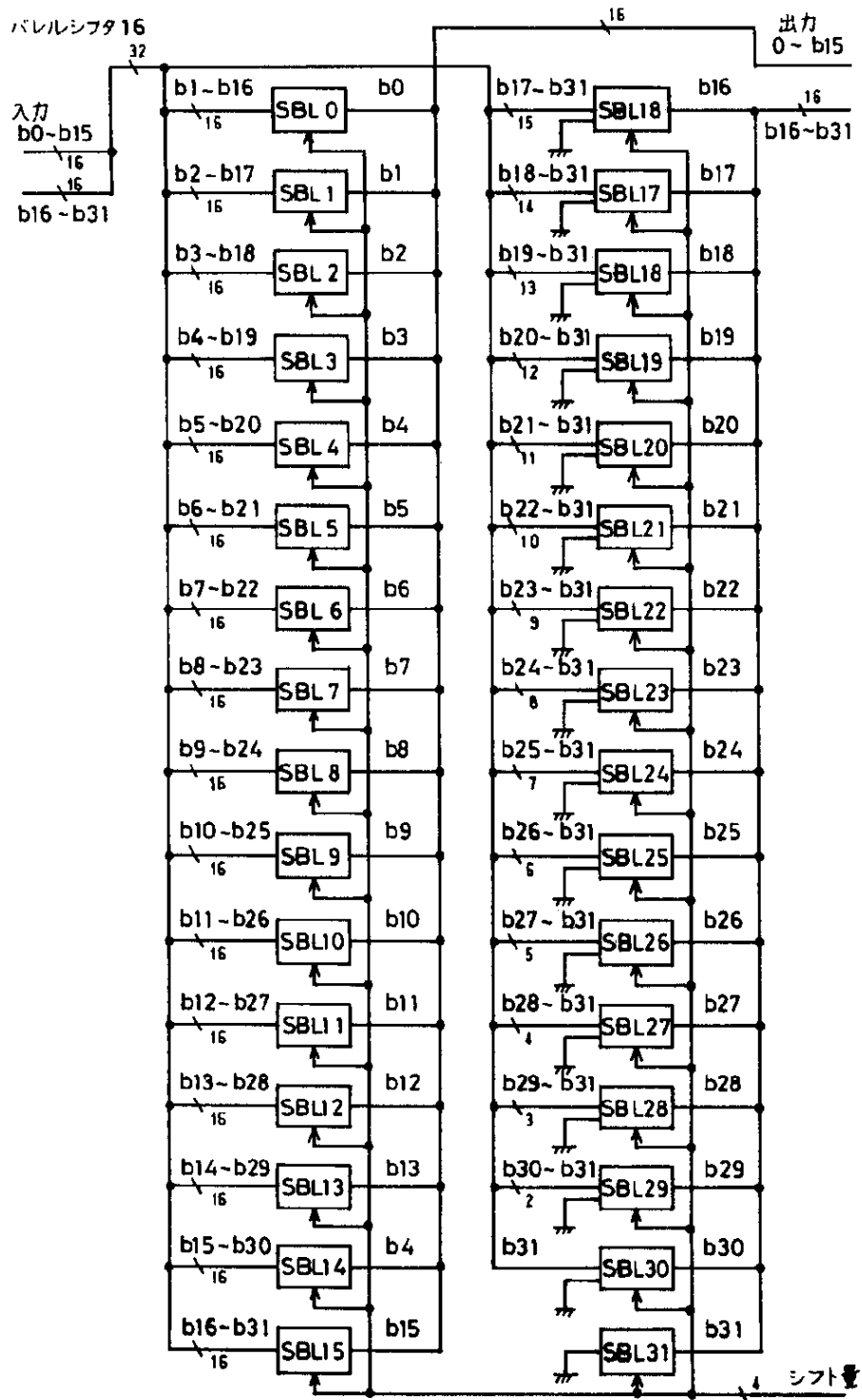
12 …第 1 レジスタ

14 …セクタ

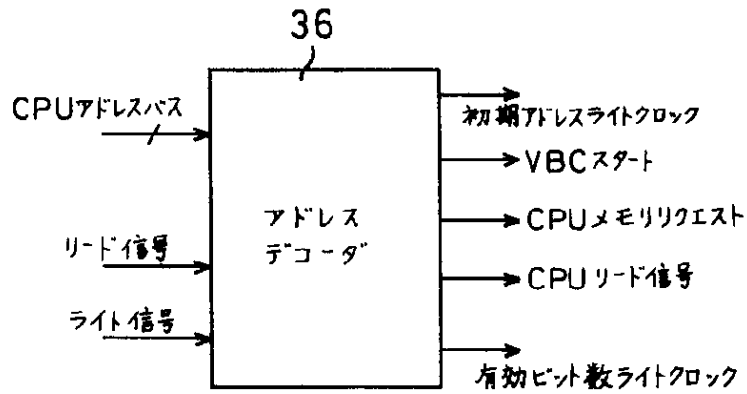
16 …バレルシフタ

20 …第 2 レジスタ

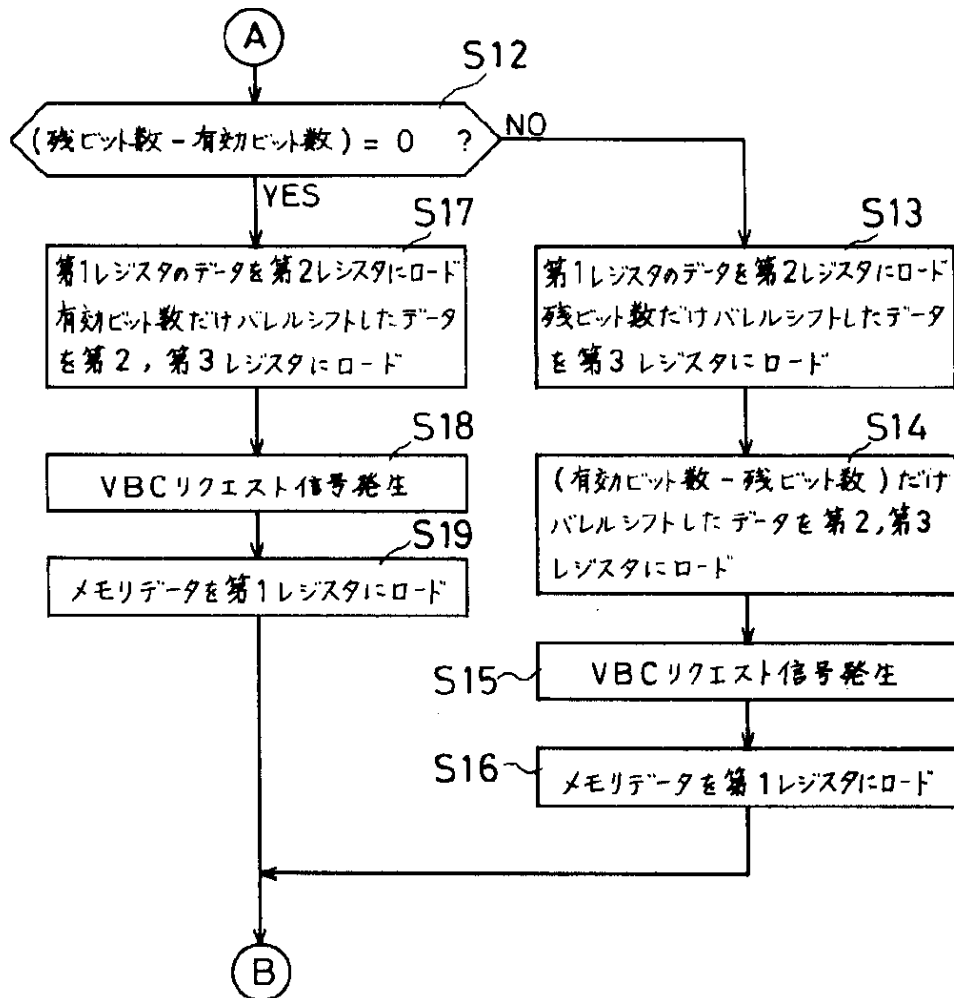
【図 2】



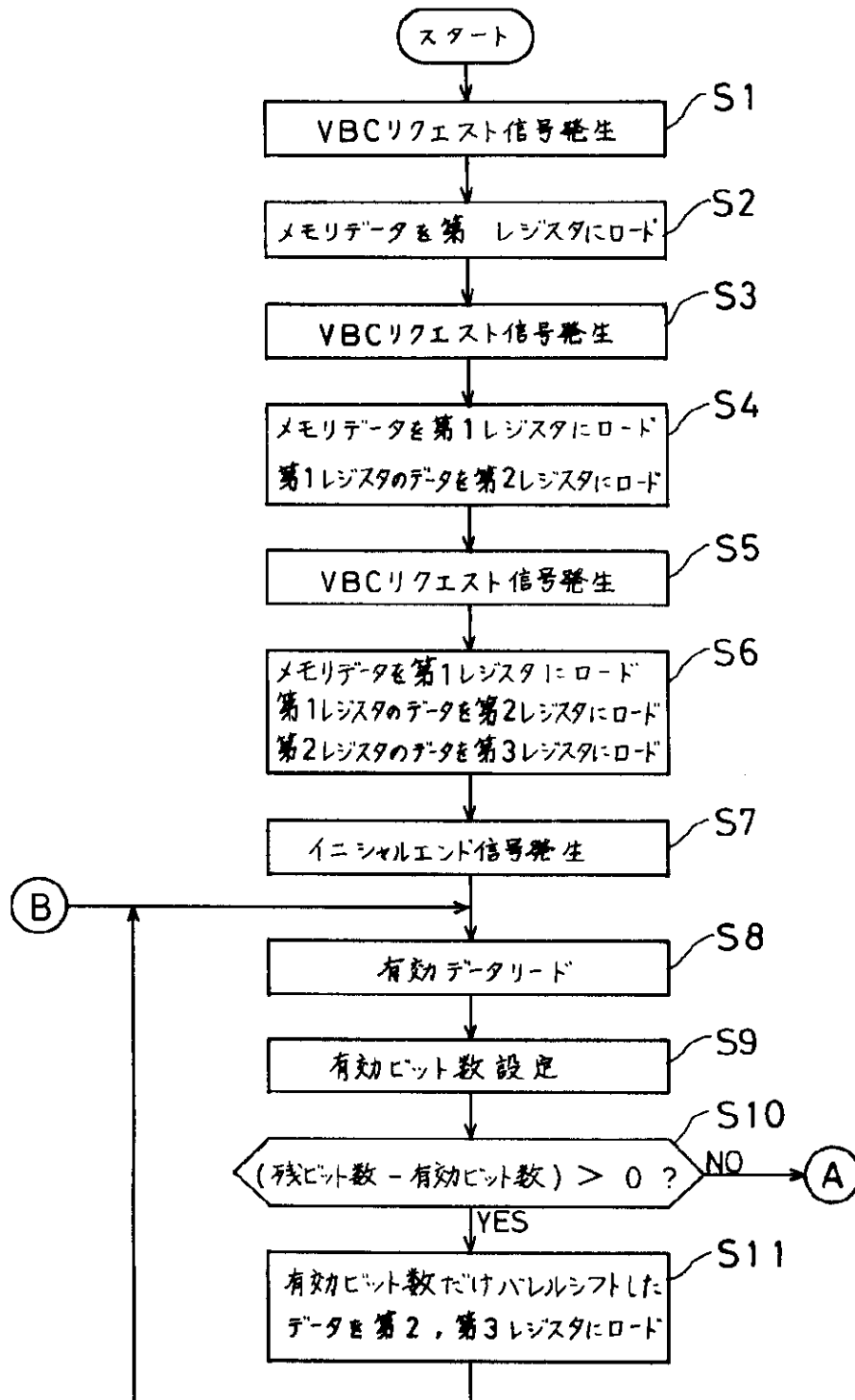
【図3】



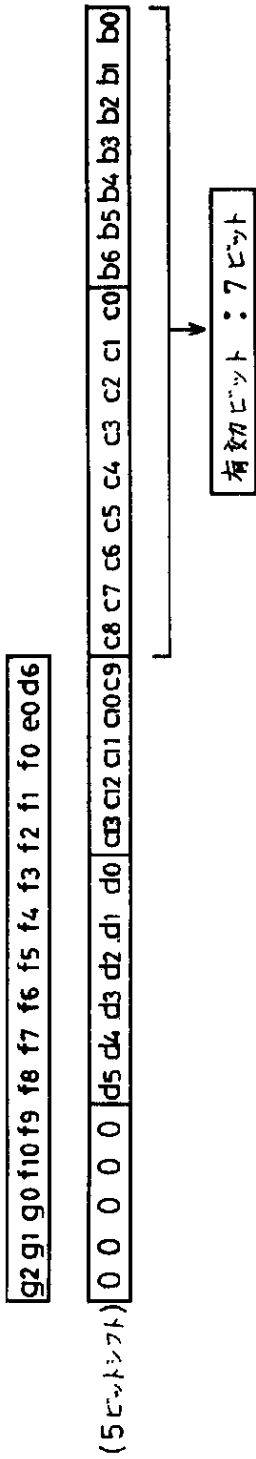
【図5】



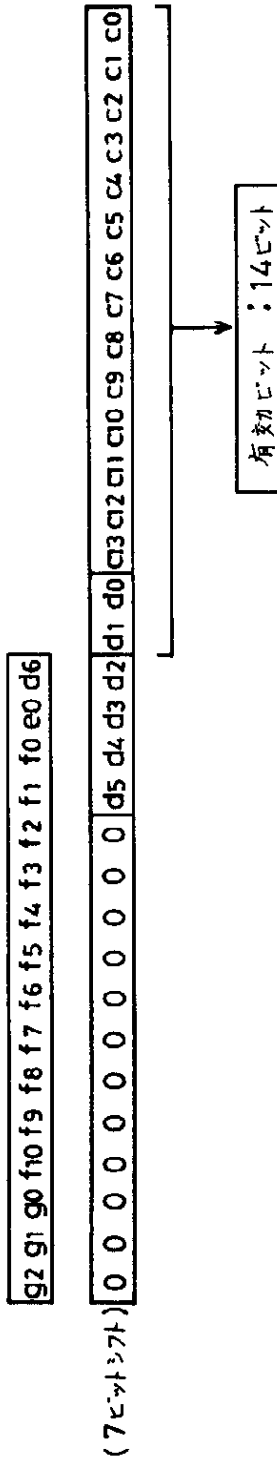
【図4】



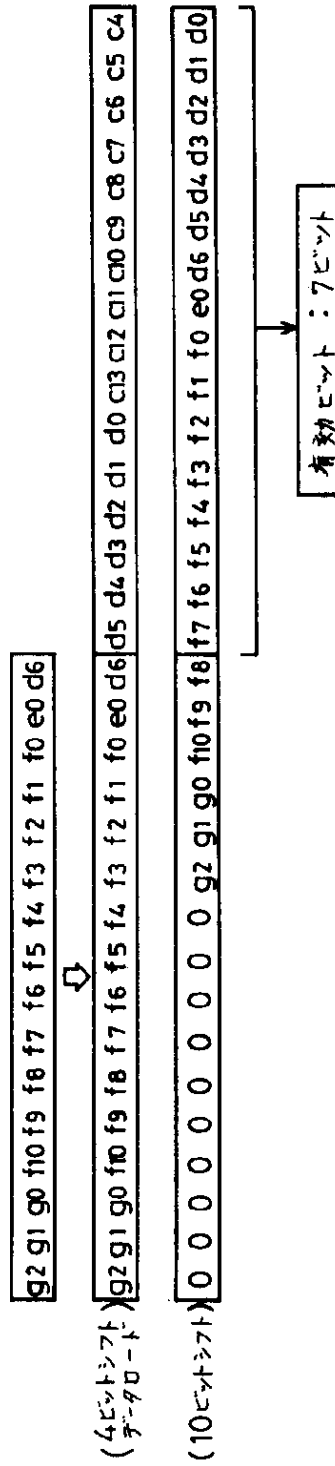
【図7】



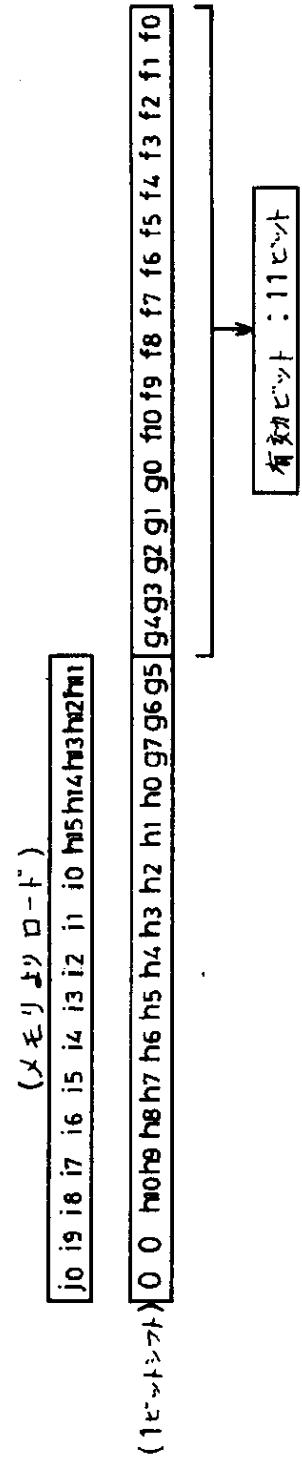
【図8】



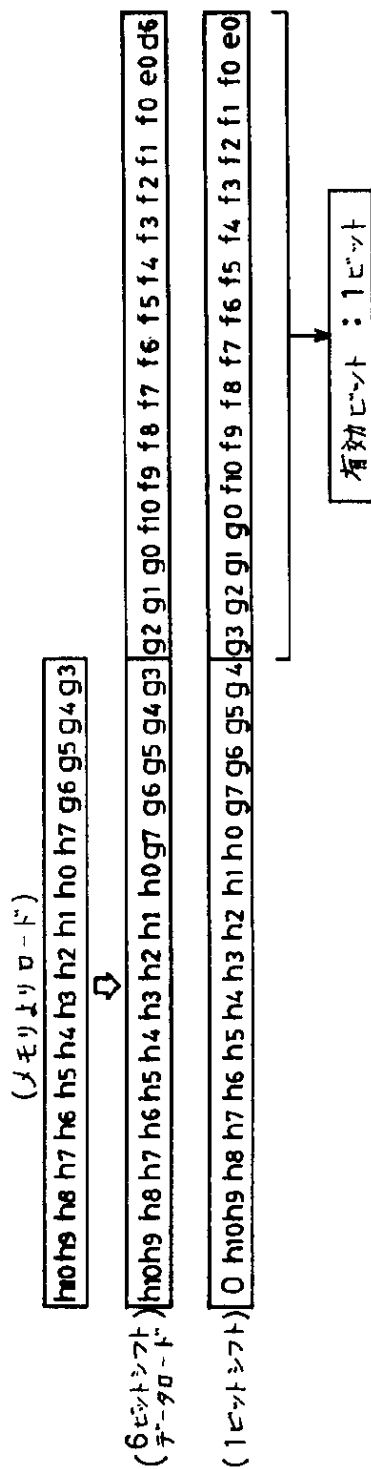
【図9】



【図11】



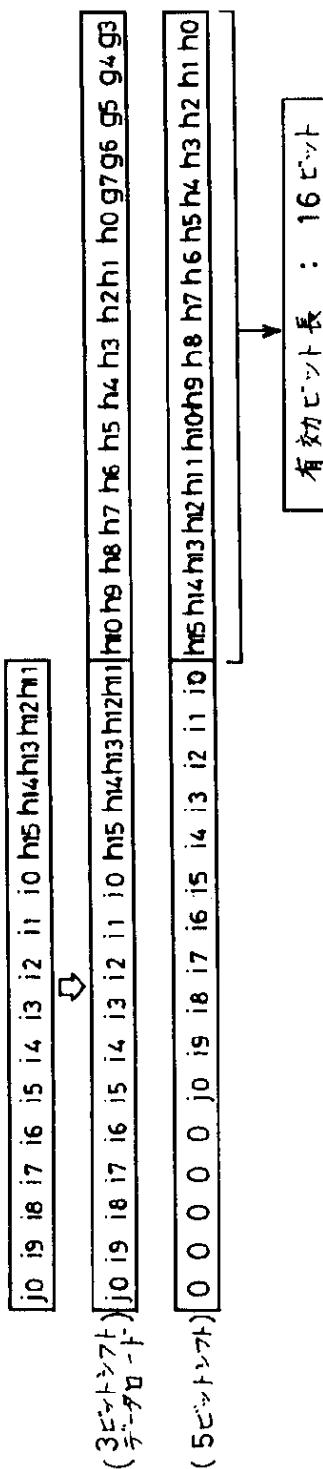
【図10】



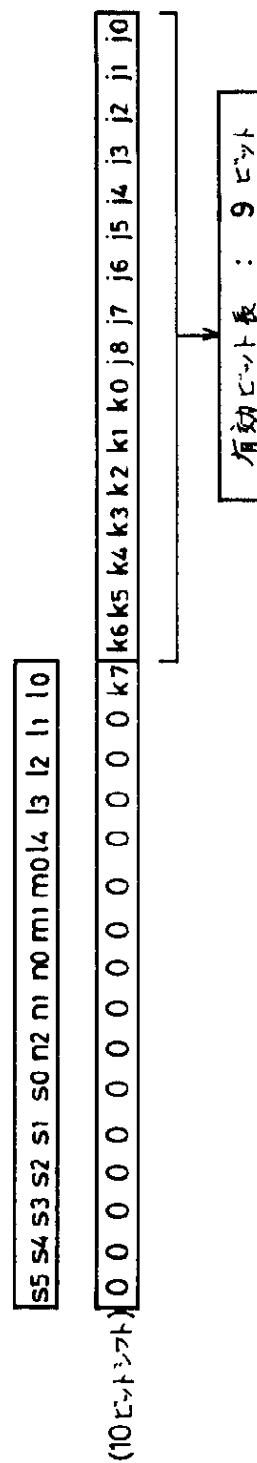
【図12】



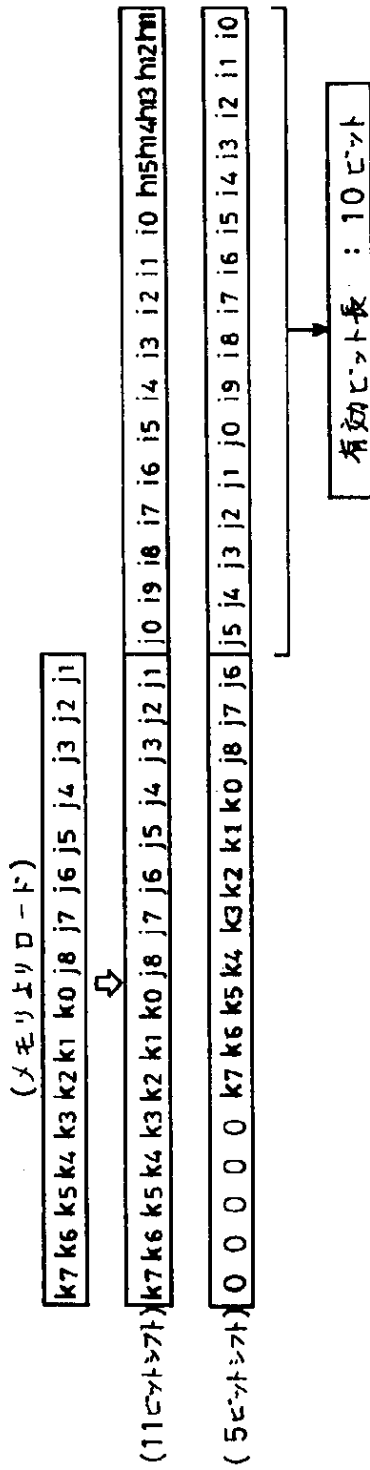
【図13】



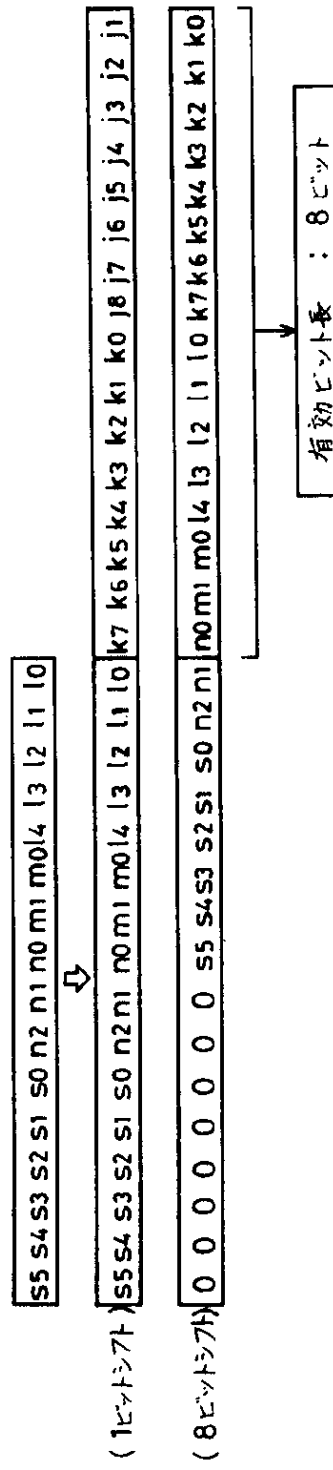
【図15】



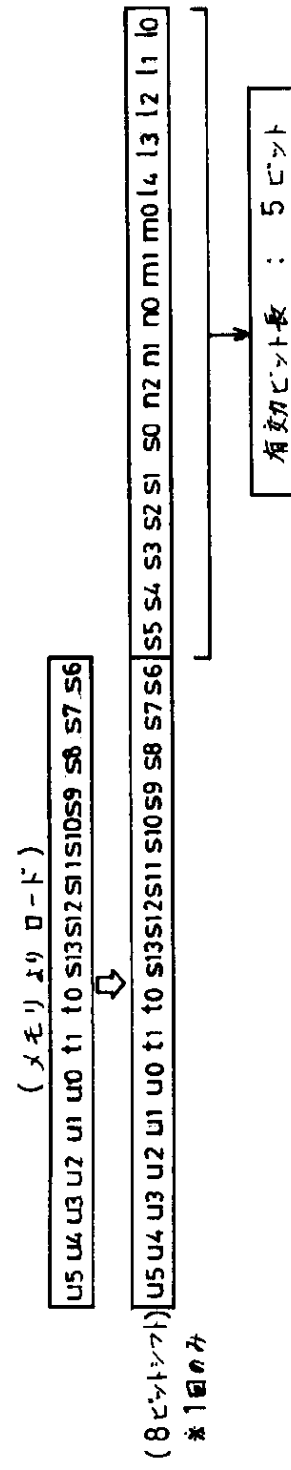
【図14】



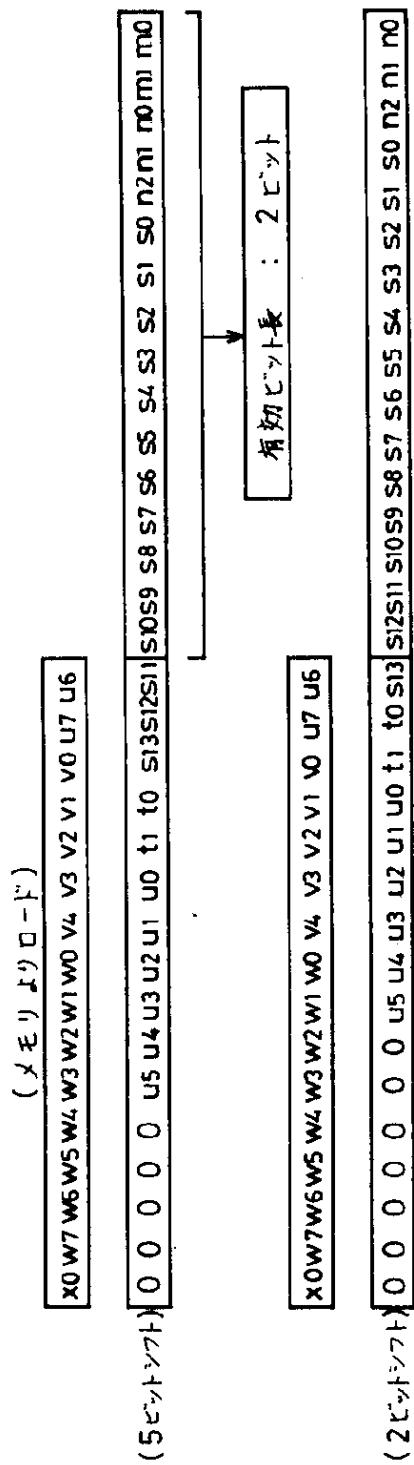
【図16】



【図17】



【図18】



フロントページの続き

(72)発明者 寺川 英明
兵庫県川西市中央町3番6号 株式会社ク
リエイティブ・デザイン内