

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-13680

(P2004-13680A)

(43) 公開日 平成16年1月15日(2004.1.15)

(51) Int. Cl.⁷

G06F 5/00

G06F 17/21

F I

G06F 5/00

G06F 17/21

テーマコード(参考)

5B009

570G

審査請求 未請求 請求項の数 6 O L (全 9 頁)

(21) 出願番号

特願2002-168638 (P2002-168638)

(22) 出願日

平成14年6月10日(2002.6.10)

(71) 出願人

000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(74) 代理人

100084250

弁理士 丸山 隆夫

(72) 発明者

小原 宏幸

東京都港区芝五丁目7番1号 日本電気株

式会社内

Fターム(参考) 5B009 SA08

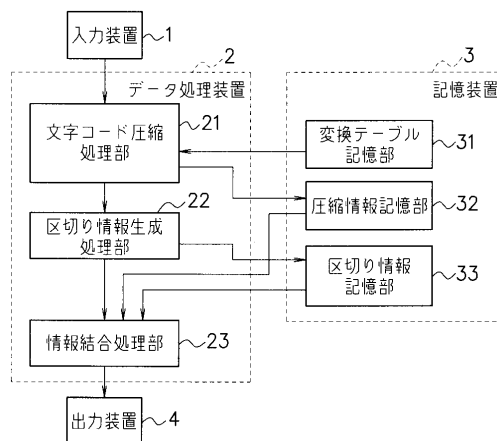
(54) 【発明の名称】 文字コード圧縮・復元装置および同方法

(57) 【要約】

【課題】 全体の記憶容量を削減する文字コード圧縮・復元装置および同方法を得る。

【解決手段】 使用頻度の高い文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、圧縮変換結果と区切り情報とをデータ処理装置2で結合する。変換テーブル記憶部31で、文字の各ビット列に対応する文字コード情報を予め記憶し、この変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す区切り情報を記憶装置3で格納する。本構成の文字コード圧縮装置において、文字の出現頻度順にビット数の少ない所に割り当てた変換テーブルを作成し、頻繁に使用される文字コードを短いビットに割り当て、全体の記憶容量を削減し、文字コードの変換効率を高める。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

入力された文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、前記圧縮変換結果と前記区切り情報とを結合するデータ処理装置と、
前記文字の各ビット列に対応する文字コード情報を予め記憶している変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す前記区切りの情報を格納する記憶装置とを有し、
前記文字の出現頻度順にビット数の少ない所に割り当てた前記変換テーブルを作成し、前記文字コードの変換効率を高めたことを特徴とする文字コード圧縮・復元装置。

【請求項 2】

前記記憶装置は、前記文字コードを変換する時に使用する変換テーブル記憶部と、前記圧縮変換した結果を格納する圧縮情報記憶部と、前記変換結果の区切り位置を示す区切り情報を格納する区切り情報記憶部とを、備えたことを特徴とする請求項 1 記載の文字コード圧縮・復元装置。

10

【請求項 3】

前記変換テーブル記憶部は、各アルファベットに対応するビット列の情報をあらかじめ記憶している記憶部であることを特徴とする請求項 2 記載の文字コード圧縮・復元装置。

【請求項 4】

入力された文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、前記圧縮変換結果と前記区切り情報とを結合するデータ処理工程と、
前記文字の各ビット列に対応する文字コード情報を予め記憶している変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す前記区切りの情報を格納する記憶工程とを有し、
前記文字の出現頻度順にビット数の少ない所に割り当てた前記変換テーブルを作成し、前記文字コードの変換効率を高めたことを特徴とする文字コード圧縮・復元方法。

20

【請求項 5】

前記記憶工程は、前記文字コードを変換する時に使用する変換テーブル記憶工程と、前記圧縮変換した結果を格納する圧縮情報記憶工程と、前記変換結果の区切り位置を示す区切り情報を格納する区切り情報記憶工程とを、備えたことを特徴とする請求項 4 記載の文字コード圧縮・復元方法。

30

【請求項 6】

前記変換テーブル記憶工程は、各アルファベットに対応するビット列の情報をあらかじめ記憶している記憶工程であることを特徴とする請求項 5 記載の文字コード圧縮・復元方法。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、文字コード圧縮・復元装置および同方法に関する。さらに詳述すると、本発明は、頻繁に使用される文字コードを短いビットに割り当てて全体の記憶容量を削減する文字コード圧縮・復元装置および同方法に関する。

40

【0002】**【従来技術】**

従来、文字コード圧縮・復元装置および同方法は、たとえば、記憶装置に適用される。従来は、入力装置から得た文字コードをそのまま記憶装置に格納したり、出力装置から外部に流したりしている。

【0003】

例えば、文字「A B C D E」の場合について、より具体的な処理手順例を以下に説明する。本文字例を、通常のビット列 (8 b i t A S C I I) で表した場合の、文字 / ビット列、の関係は以下となる。

A / 0 1 0 0 0 0 0 1、B / 0 1 0 0 0 0 1 0、C / 0 1 0 0 0 0 1 1、

50

D / 0 1 0 0 0 1 0 0、E / 0 1 0 0 0 1 0 1

【0004】

上記の従来例によれば、下記の特徴がある。

1) 1文字を常に8bitで表しているため、文字の区切りを示す必要がない。

2) 必要なビット数は、8bit×5文字=40bit、である。

【0005】

本発明と技術分野の類似する先願発明例1として、特開平11-85459号公報の「文字データ符号化方法および記録媒体」がある。本先願発明例1では、基本的にはASCIIコード表に従って、1文字を1つのコードに符号化するが、複数文字からなる特定のキーワードについては、制御文字用に用意された領域コードを用いて符号化する。即ち、文字数とその出現頻度とを考慮して、データ圧縮効果の高いものから順に選択された16個のキーワードを、コード10H~1fHに符号化する。これにより、より少ない記憶容量にて効率よく記憶媒体に格納可能なように文字データを符号化している。

10

【0006】

先願発明例2として、特開平11-55125号公報の「文字データの圧縮・復元方法」がある。本先願発明例2では、原データを判別して内部番号列に変換して符号化する。また、内部番号に変換された文字列が辞書に保持されていない場合には、2つのグループに分け、ひらがな、漢字等の文字を符号化し、グループを表すビットを付して出力する。これにより、小さいデータでも高い圧縮率を得ることができる、としている。

【0007】

先願発明例3として、特開平9-69785号公報の「データ圧縮方法及びデータ圧縮装置」がある。本先願発明例3では、特定文字のコードデータとその圧縮コードデータとを対応付けた対応一覧表を用意しておき、圧縮対象データから取り出した1文字分の文字コードデータがその対応一覧表に存在するか否かを検索する。その結果、当該データが対応する一覧表に存在する場合には、その圧縮コードデータを対応一覧表から読み出し、存在しない場合には、当該データをそのまま出力する。これにより、効果的なデータ圧縮を行うとしている。

20

【0008】

【発明が解決しようとする課題】

しかしながら、従来文字コード圧縮・復元装置および同方法では、入力装置から得た文字コードをそのまま記憶装置に格納している。このため、1文字当たり常に1バイト(8ビット)の容量を使用しており、文章全体に渡って無駄なビットが記憶容量の多くを占めているという問題を伴う。

30

【0009】

本発明は、頻繁に使用される文字コードを短いビットに割り当て、全体の記憶容量を削減する文字コード圧縮・復元装置および同方法を提供することを目的とする。

【0010】

【課題を解決するための手段】

かかる目的を達成するため、本発明の文字コード圧縮・復元装置は、入力された文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、圧縮変換結果と区切り情報とを結合するデータ処理装置と、文字の各ビット列に対応する文字コード情報を予め記憶している変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す区切りの情報を格納する記憶装置とを有し、文字の出現頻度順にビット数の少ない所に割り当てた変換テーブルを作成し、文字コードの変換効率を高めたことを特徴としている。

40

【0011】

また、上記の記憶装置は、文字コードを変換する時に使用する変換テーブル記憶部と、圧縮変換した結果を格納する圧縮情報記憶部と、変換結果の区切り位置を示す区切り情報を格納する区切り情報記憶部とを備え、変換テーブル記憶部は、各アルファベットに対応するビット列の情報をあらかじめ記憶している記憶部とする。

50

【0012】

本発明の文字コード圧縮・復元方法は、入力された文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、圧縮変換結果と区切り情報とを結合するデータ処理工程と、文字の各ビット列に対応する文字コード情報を予め記憶している変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す区切りの情報を格納する記憶工程とを有し、文字の出現頻度順にビット数の少ない所に割り当てた変換テーブルを作成し、文字コードの変換効率を高めたことを特徴としている。

【0013】

また、上記の記憶工程は、文字コードを変換する時に使用する変換テーブル記憶工程と、圧縮変換した結果を格納する圧縮情報記憶工程と、変換結果の区切り位置を示す区切り情報を格納する区切り情報記憶工程とを備え、変換テーブル記憶工程は、各アルファベットに対応するビット列の情報をあらかじめ記憶している記憶工程とするとよい。

10

【0014】

【発明の実施の形態】

次に、添付図面を参照して本発明による文字コード圧縮・復元装置および同方法の実施の形態を詳細に説明する。図1から図8を参照すると、本発明の文字コード圧縮・復元装置および同方法の一実施形態が示されている。

【0015】

本発明の文字コード圧縮・復元装置および同方法は、アルファベットとして使用されている文字コードが有しているビットサイズを減少することで、文章などで使用している文字列全体のビット数の削減を図り、全体として使用するデータの容量を圧縮するものである。また、可逆的な復元方式を用いることで、圧縮した情報から、元の文字コードへの復元動作も可能とする。

20

【0016】

更に、1文字ずつ順番に圧縮・復元できる方式の為、先頭から1文字ずつ入力される場合や末尾の文字を1文字削除するなどといった場合でも、その影響範囲を該当文字の部分だけに押さえ込むことを可能とする。本構成の内容を以下に詳述する。

【0017】

(構成例)

図1および図2は、本発明の文字コード圧縮・復元装置および同方法の実施形態に適用される文字コード圧縮・復元装置の構成例を示すブロック図である。

30

図1を参照すると、文字コード圧縮装置であり、文字コード圧縮時に適用される機能部を示している。本機能部は、キーボードなどの入力装置1と、プログラム制御により動作するデータ処理装置2と、情報を記憶する記憶装置3と、情報を外部に取り出す為の出力装置4とを含む。

【0018】

データ処理装置2は、入力装置1より入力された文字コードを圧縮する文字コード圧縮処理部21と、文字コードの区切りの情報を生成する区切り情報生成処理部22と、外部に出力する際に変換結果と区切り情報を結合する情報結合処理部23とを備えている。

【0019】

記憶装置3は、文字コードを変換する時に使用する変換テーブル記憶部31と、圧縮変換した結果を格納する圧縮情報記憶部32と、変換結果の区切り位置を示す区切り情報を格納する区切り情報記憶部33とを備えている。変換テーブル記憶部31は、各アルファベットに対応するビット列の情報を、あらかじめ記憶している。

40

【0020】

次に、図2を参照すると、本実施例は、文字コードを復元する時のものであり、先の図1にて圧縮した情報を外部から入力する為の入力装置1と、プログラム制御により動作するデータ処理装置5と、情報を記憶する記憶装置6と、情報を外部に取り出す為の出力装置4とを含む。

【0021】

50

記憶装置 6 は、文字コードの圧縮情報部分を記憶する圧縮情報記憶部 6 1 と、区切り情報部分を記憶する区切り情報記憶部 6 2 と、文字コードを復元する時に使用する変換テーブル記憶部 6 3 とを備える。なお、この変換テーブル記憶部 6 3 は、各ビット列に対応する文字コードの情報を、あらかじめ記憶している。

【0022】

データ処理装置 5 は、入力装置 1 から得た情報を、文字コードの圧縮情報と区切り情報に分離する入力データ分離処理部 5 1 と、記憶装置 6 の情報を元の文字コードに復元する文字コード復元処理部 5 2 とを備える。

【0023】

(動作例)

次に、図 1 ~ 図 7 を参照して、本実施例の動作について詳細に説明する。なお、図 3 は、記憶装置 3 および記憶装置 6 に記憶された変換テーブルの構成例を示す。図 4 は、文字列、文字のビット列、および文字の区切りビット列の構成例を示す。図 5 は、文字列例に対応する圧縮ビット列、区切りビット列、および従来のコード例を表した図である。図 6 および図 7 は、処理手順例を示すフローチャートである。

【0024】

これらの図において、図 1 および図 6 は、文字コードから圧縮データへ圧縮する処理例を、図 2 および図 7 では、圧縮されたデータから元の文字コードへの復元を行う処理例を表している。

【0025】

まず、文字コードを入力する(ステップ A 1)。入力が未終了の場合は(ステップ A 2 / NO)、入力文字を圧縮ビット列に変換し(ステップ A 3)、変換後のビット列から区切りビット列を生成する(ステップ A 4)。

入力が終了した場合は(ステップ A 2 / YES)、圧縮ビット列と区切りビット列の結合を実行し(ステップ A 5)、実行後の結合されたビット列を外部へ出力する(ステップ A 6)。

【0026】

図 1 の入力装置 1 から与えられた文字コードは(ステップ A 1)、データ処理装置 2 により記憶装置 3 を参照して、別のビット列に置き換える(ステップ A 3、A 4、A 5)。記憶装置 3 は、図 3 に示す通り、各文字に対応したビット列を保持し、それをテーブルとして記憶している。

【0027】

データ処理装置 2 は、入力された文字コードを、記憶装置 3 を元にしたビット列に置き換える(ステップ A 3、A 4、A 5)。その際、データ処理装置 2 では、文字コードを置き換えたビット列の区切り情報を記憶するために、文字の区切り部分識別のためのビット列を生成する(ステップ A 4)。

【0028】

図 4 は、文字列“ABCDE”と、文字ビット列と、文字の区切りビット列とを示している。本図 4 に示す通り、置き換えたビット列の先頭ビットは“1”、残りのビットは“0”に相当するビット列を生成する。これにより、区切りビット列のビットが“1”の所に該当する圧縮ビット列のビットが、圧縮した文字コードの先頭ビットとなり、続く“0”のビット列の部分が、圧縮した文字コードの残りの部分となる。

【0029】

図 6 に示す通り、上記の処理を入力が終了するまで行い、入力が終了した時点で、データ処理装置 2 にて、圧縮ビット列と区切りビット列の結合を行い、出力装置 4 にて出力を行う。

【0030】

次に、図 2 の入力装置 1 から与えられたデータは、データ処理装置 5 の入力データ分離処理部 5 1 で、圧縮情報データと区切り情報データの 2 つに分割を行う。分割したこれらのデータはそれぞれ、圧縮情報記憶部 6 1 と区切り情報記憶部 6 2 に格納する。次に、デー

10

20

30

40

50

タ処理装置 5 の文字コード復元処理部 5 2 では、記憶装置 6 の圧縮情報記憶部 6 1 と区切り情報記憶部 6 2 と変換テーブル記憶部 6 3 のデータを、元の文字コードへの復元処理を行う。データ処理装置 5 は、区切り情報記憶部 6 2 から文字の区切り情報として、ビット“1”とそれに続くビット“0”を、次にビット“1”が現れるか、区切りデータが末尾になるまで順に取り出す。その時に、何ビット取り出したかをカウントしておく。

【0031】

次に、文字コード復元処理部 5 2 は、先ほどカウントした数の分だけ、圧縮情報記憶部 6 1 から圧縮情報のビットを取り出す。次に、文字コード復元処理部 5 2 は、取り出した圧縮情報のビットを、変換テーブル記憶部 6 3 のテーブルで検索し、該当するビット列に相当する文字コードを入手する。

10

【0032】

図 3 に示す通り、変換テーブル記憶部 6 3 が記憶する変換テーブルは、文字コードとビット列のテーブルとなっており、ビット列を順に検索することで、そのビット列に相当する文字コードを割り出すことが出来る。取り出した文字コードは、出力装置 4 により外部に取り出す。

【0033】

図 7 に示す通り、先ずデータを入力する（ステップ B 1）。入力したデータを、圧縮ビット列と区切りビット列に分離する（ステップ B 2）。区切りビット列が残っている場合は（ステップ B 3 / YES）、区切りビット列を元に、圧縮ビット列から 1 文字分のビット列を入手し（ステップ B 4）、1 文字分のビット列から、テーブルを元に文字コードを入手する（ステップ B 5）。圧縮ビット列から 1 文字分のビット列入手（ステップ B 4）と、文字コード入手（ステップ B 5）は、区切りビット列が無くなるまで繰り返し実行される（ステップ B 3 / NO）。

20

【0034】

このように、区切り情報記憶部 6 2 に区切り情報が残っている場合には（ステップ B 3 / YES）、再度、区切り情報記憶部 6 2 から取り出した情報を元に、圧縮情報記憶部 6 1 から圧縮情報を取り出し（ステップ B 4）、変換テーブル記憶部 6 3 のテーブルから文字コードを割り出す（ステップ B 5）。

1 文字分のビット列から入手した文字コードは、出力装置 4 により、外部に取り出すものとする（ステップ B 6）。

30

【0035】

図 8 の変換テーブルによるビット列で表した場合の、文字 / ビット列、の関係は、以下となる。

A / 1 0、 B / 0 1 1 1、 C / 0 1 0 1、
D / 0 1 1、 E / 1

上記の具体例では、1 文字を可変長で表しているため、文字の区切りの位置を示す必要がある。そのため、区切り識別用のビット列を設けることとする。

【0036】

例えば、区切りの識別子には、先頭ビットが“1”、残りのビットが“0”としたものを使用する。一例を、以下に示す。以下の、文字 / ビット列、の関係の具体例は、図 8 のテーブルを使用した場合に該当している。

40

3) 文字列； A B C D E
4) 文字のビット列； 1 0 0 1 1 1 0 1 0 1 0 1 1 1 1 4 b i t
5) 区切り用ビット列； 1 0 1 0 0 0 1 0 0 0 1 0 0 1 1 4 b i t

上に掲げた具体例では、合計が 2 8 b i t となる。この場合のビット数を上述した従来の処理例と比較すると、本例 / 従来例が $28 / 40 = 70\%$ となる。

【0037】

（実施例の効果）

第 1 の効果は、通常 8 ビットで表現される文字コードを、最低 2 ビット ~ 最悪 8 ビットで表わすことで文章全体で使用する容量を削減でき、少ないメモリで多くの文字が記憶でき

50

る。また、ネットワークなどで送受信する際にも、流すデータ量が削減され、転送速度の向上とトラフィックの軽減がなされる。

【0038】

なお、図4に示した通り、単純な「ABCDE」といった文字列の場合には、70%の圧縮率となっている。しかし、通常の文書中に頻繁に使用される文字を割り出し、その文字を頻度順にビット数の少ない所に割り当てるようなテーブルを作成することで、更に圧縮率を高めることが可能となる。

【0039】

(他の実施例)

図1の変換テーブル記憶部31のテーブルの内容を、当事者間で了解され、外部に知らせていない独自のテーブルを使用することで、入出力データが外部に漏洩した際にも、データが暗号化された状態となり、機密保持を行うことが出来る。 10

【0040】

なお、上述の実施形態は本発明の好適な実施の一例である。ただし、これに限定されるものではなく、本発明の要旨を逸脱しない範囲内において種々変形実施が可能である。

【0041】

【発明の効果】

以上の説明より明らかなように、本発明の文字コード圧縮・復元装置および同方法は、入力された文字の文字コードを圧縮変換し、該文字コードの区切りの情報を生成し、圧縮変換結果と区切り情報とを結合し、文字の各ビット列に対応する文字コード情報を予め記憶している。また、変換テーブルを使用して変換された文字コード、および該変換結果の区切り位置を示す区切りの情報を格納し、文字の出現頻度順にビット数の少ない所に割り当てた変換テーブルを作成して、文字コードの変換効率を高めている。 20

【図面の簡単な説明】

【図1】本発明の文字コード圧縮・復元装置の実施形態に適用され、文字コード圧縮時に使用される機能部の構成例を示すブロック図である。

【図2】本発明の文字コード圧縮・復元装置の実施形態に適用され、文字コード復元時に使用される機能部の構成例を示すブロック図である。

【図3】記憶装置3および記憶装置6に記憶された変換テーブルの構成例を示す。

【図4】文字列、文字のビット列、および文字の区切りビット列の構成例を示す。 30

【図5】文字列例に対応する圧縮ビット列、区切りビット列、および従来のコード例を表した図である。

【図6】処理手順例を示す第1のフローチャートである。

【図7】処理手順例を示す第2のフローチャートである。

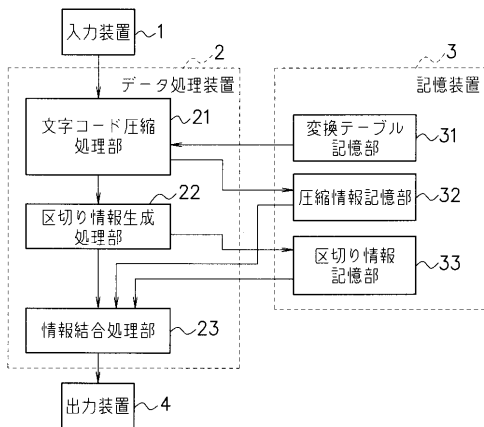
【図8】変換テーブルの構成例を示している。

【符号の説明】

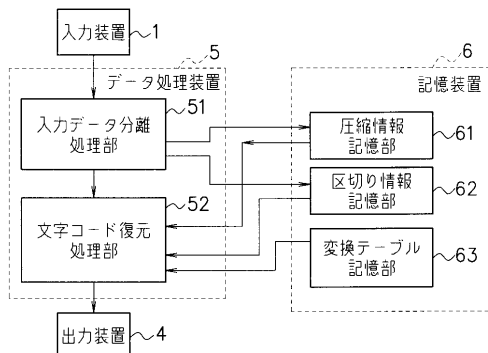
- 1 入力装置
 - 2 データ処理装置
 - 3 記憶装置
 - 4 出力装置
 - 5 データ処理装置
 - 6 記憶装置
 - 21 文字コード圧縮処理部
 - 22 区切り情報生成処理部
 - 23 情報結合処理部
 - 31 変換テーブル記憶部
 - 32 圧縮情報記憶部
 - 33 区切情報記憶部
 - 51 入力データ分離処理部
 - 52 文字コード復元処理部
- 40
- 50

- 6 1 圧縮情報記憶部
- 6 2 区切り情報記憶部
- 6 3 変換テーブル記憶部

【 図 1 】



【 図 2 】



【 図 3 】

記憶装置3および記憶装置6(変換テーブル)の例

文字	ビット列
E	1
T	0

I	11
A	10
N	01
M	00

S	111
D	011
U	110
R	101

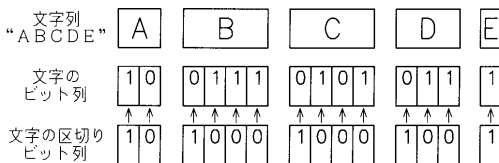
K	010
G	001
W	100
O	000

H	1111
B	0111
V	1110
F	1101
L	1011

Z	0011
C	0101
P	1001
X	0110

J	1000
Y	0100
Q	0010

【 図 4 】

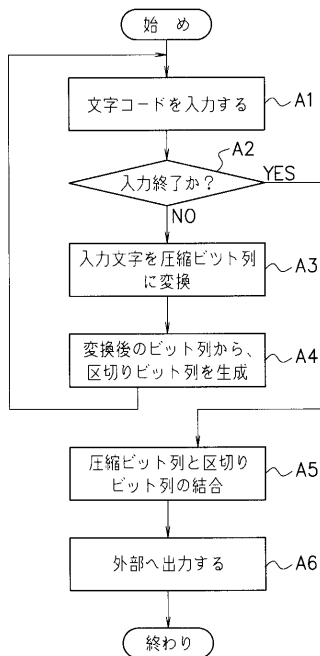


【 図 5 】

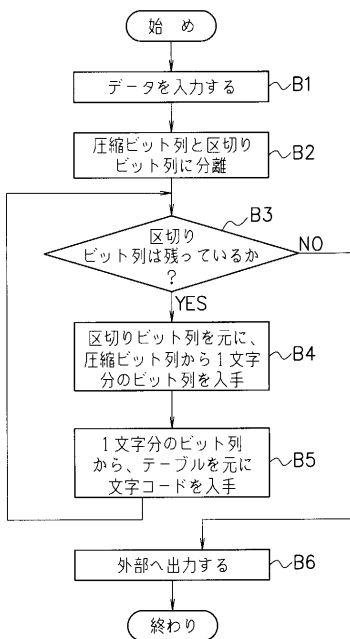
文字列例) ABCDE	圧縮ビット列 (図3のテーブル を使用)	区切りビット列 先頭ビット=1 その他ビット=0	従来のコード (ASCIIコード)
A	1 0	1 0	01000001
B	0 1 1	1 0 0	01000010
C	0 1 0 1	1 0 0 0	01000011
D	0 1 1	1 0 0	01000100
E	1 1	1 1	01000101
	14bit	14bit	40bit

$(14+14)/40=70\%$

【 図 6 】



【 図 7 】



【 図 8 】

E...1	H...1111
T...0	B...0111
	V...1110
I...11	F...1101
A...10	L...1011
N...01	
M...00	Z...0011
	C...0101
S...111	P...1001
D...011	X...0110
U...110	
R...101	J...1000
	Y...0100
K...010	Q...0010
G...001	
W...100	
O...000	