

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

H03M 7/30

H03M 7/42



[12] 发 明 专 利 说 明 书

[21] ZL 专利号 95109053.4

[45] 授权公告日 2003 年 11 月 5 日

[11] 授权公告号 CN 1127219C

[22] 申请日 1995.7.31 [21] 申请号 95109053.4

[30] 优先权

[32] 1994. 8. 1 [33] US [31] 283346

[71] 专利权人 开放电视公司

地址 美国加利福尼亚州

[72] 发明人 P·罗哈吉

审查员 张 军

[74] 专利代理机构 中国专利代理(香港)有限公司

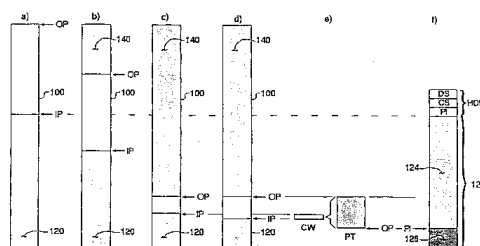
代理人 马铁良 张志醒

权利要求书 3 页 说明书 11 页 附图 4 页

[54] 发明名称 压缩和去压缩数据文件的方法

[57] 摘要

一种对调节的单调压缩文件就地去压缩的方法，该压缩文件具有压缩码字的第一部分和简单正文的第二部分，形成的去压缩文件具有一定长度并包括简单正文。步骤是：指定一存贮缓冲器 100 具有去压缩文件的大小，把调节的单调压缩文件 120' 存入缓冲器底部 120，对压缩文件中第一部分的压缩码字去压缩，在缓冲器顶部 140 产生去压缩文件的相应部分的简单正文。该方法还可把源文件压缩为含有压缩码字的压缩文件。



1. 一种对调节的单调压缩文件进行就地去压缩的方法，所述压缩文件具有包含压缩码字的第一部分和包含简单正文的第二部分，去压缩后的文件具有长度大小，并包含简单正文，其特征在于，所述方法包括如下步骤：

指定一个具有去压缩文件长度大小的存贮缓冲器；
将所述的调节的单调压缩的文件存入所述缓冲器的底部；和
对调节的单调压缩文件第一部分中的压缩码字去压缩从而产生缓冲器
10 顶部去压缩文件的相应部分的简单正文。

2. 根据权利要求1的方法，其特征在于，所述的去压缩步骤包括：
对缓冲器中某一位置设置一输入指针，该位置相应于调节的单调压缩文件的起点；
对所述缓冲器的起点设置一输出指针；
15 从输入指针指向的所述缓冲器的所述位置读取码字并前行输入指针；
解码所述码字去产生简单正文；
把所述简单正文写入所述输出指针指向的所述缓冲器中的所述位置，
并前行输出指针；
重复所述读取、解码和写入步骤，直到所述输出指针相应于所述第二
20 部分的所述位置。

3. 根据权利要求1的方法，其特征在于，
所述调节的单调压缩文件还具有一定长度大小，并包括表示该调节的单调压缩文件长度大小、去压缩文件的长度大小和所述第二部分的位置的数据；

- 25 所述指定步骤包括读取报头的步骤以确定去压缩文件的长度大小；所述存入步骤包括如下步骤：

读取所述报头以确定调节的单调压缩文件的长度大小和压缩文件的长度大小；

- 30 从调节的单调压缩文件的大小和去压缩文件的大小计算调节的单调压缩文件的起点在缓冲器中的位置；和

把调节的单调压缩文件以计算的位置为起点存入缓冲器中;

所述去压缩步骤包括如下步骤:

读取所述报头以确定所述第二部分的所述位置;

对调节的单调压缩文件的起点在所述缓冲器中的计算位置设置一输入

5 指针;

对所述缓冲器的起点设置一输出指针;

从所述缓冲器中输出指针指向的位置读取码字并使输入指针前行;

对所述码字解码以产生简单正文;

把所述简单正文写入缓冲器中输出指针指向的位置,并使输出指针前

10 行;

重复所述读取、解码和写入步骤,直到输出指针相应于所述第二部分的所述位置。

4. 根据权利要求1的方法,其特征在于,

15 在所述第一部分中的所述压缩码字包括一代表了重复串中的字符位置
和数目的码字,所述重复串使用了重叠匹配;和

所述去压缩步骤包括对所述重复串去压缩的步骤,所述重复串表示码字,通过从重复串的去压缩文件的所述位置到去压缩文件的相应位置、一次一个地复制重复串中的字符数完成去压缩过程。

20 5. 根据权利要求1的方法,其特征在于,所述压缩码字的长度是半字节的整数倍。

6. 一种把具有一定长度大小的源文件压缩为调节的单调压缩文件的方法,其特征在于,包括如下步骤:

把源文件单调压缩为含有压缩码字的压缩文件;

25 调节压缩文件以产生调节的单调压缩文件,该单调的压缩文件具有含有压缩码字的第一部分和含有简单正文的第二部分。

7. 根据权利要求6的方法,其特征在于,所述压缩码字的长度是半字节的整数倍。

8. 根据权利要求6的方法,其特征在于,所述调节步骤包括以下步骤:

30 模拟指定一具有源文件长度大小的存贮缓冲器,

- 把压缩文件模拟存贮到缓冲器的底部;
设置一输入指针指向压缩文件的起点;
模拟设置一输出指针指向缓冲器的起点;
模拟去压缩压缩文件, 该步骤如下:
- 5 从压缩文件中读取输入指针指向的压缩码字, 并使输入指针前行;
模拟解码所述码字以产生简单正文;
把所述简单正文模拟写入到所述缓冲器中输出指针指向的位置, 并使
输出指针前行;
- 重复所述读取、模拟解码和模拟写入步骤, 直到解码中模拟解码的码
10 字使输出指针追上输入指针; 和
- 当模拟解码码字使输出指针追上输入指针时, 相应于输出指针指向的
源文件的位置从源文件中复制简单正文以产生调节的单调压缩文件。
9. 根据权利要求 6 的方法, 其特征在于, 还包括把报头加到所述调
节的单调压缩文件上的步骤, 所述报头含有代表源文件大小、调节的单调
15 压缩文件的大小和在含有简单正文的调节的单调压缩文件中的位置的数
据。
10. 根据权利要求 6 的方法, 其特征在于, 所述调节的单调压缩步骤
使用了重叠匹配以产生压缩码字。

压缩和去压缩数据文件的方法

5 技术领域

本发明涉及一种压缩系统数据文件和去压缩系统数据文件的方法，在该系统中，压缩以离线完成，去压缩以实时完成，这种系统是存储器限制的系统，例如音频视频相互作用系统。

背景技术

- 10 音频视频相互作用系统（AVI）是一种广播相互作用系统，通过该系统，观众可以和电视节目相互作用。在这种系统中，每个接收现场包括一个处理器，用来接收来自发射现场的以 AVI 信号传送的 AVI 应用程序。AVI 信号是包括音频分量和视频分量的组合信号，类似于标准电视信号；还包括相互作用分量，载有实现应用程序的码数据模块。应用节目编辑利用标
- 15 准的程序编译设备（例如编译程序）离线地产生码和数据模块，这些码和数据模块然后被处理为相互作用分量，相互作用分量与音频和视频分量组合形成 AVI 信号。

- AVI 组合信号被发送到接收现场。从 AVI 组合信号中提取音频和视频分量进行处理，并以周知方式（例如在常规电视系统中）产生声音和图象，
- 20 提取相互作用分量，并将码和数据模块放入接收现场的处理器存储器中。然后，处理器执行码模块，并处理数据模块，产生将分别加到 AVI 音频声音和视频图象上的声音和图形图象，并根据观众的输入而与电视节目相互作用。

- 显然，接收现场的设备的生产和销售成本必须要比较低。减少每个处
- 25 理器中的存储器的数量，以及使用较低计算能力的微处理器（如 SGS/Thomson S79）可以降低成本。

- 再者，应用程序将被分为多个小的码和数据模块（典型地小于 32KB）。为了优化适合于相互作用分量的带宽，码和数据模块首先被压缩，然后再发送。因为模块很小，且压缩要离线进行。所以压缩速度无关紧要。另一
- 30 方面，去压缩速度必须很快，即使对于计算能力很低的微处理器，例如

SGS/Thomson ST9（微处理器不适合去压缩，因为其比特移位能力有限，例如，只有一个比特移位）。此外，在接收现场处理器只有较少量的 RAM 用来存贮和执行应用程序。因此，所使用的任何去压缩方法都要求使用尽量少的存储器。

- 5 先有技术的压缩技术包括 Lempel-Zir'77 系列压缩算法。这些压缩技术的基本思路是在源数据中经常存在重复。通过用一包括有指向先前出现的内容（可能话）和匹配长度的指针的特定码字来替换重复的内容（称为“返回指针”），例如：源串“abcdabcf”可以如下方式编号：

Copy characters 'a', 'b', and 'c' to output;

- 10 Copy a match of Size 3, starting 4 bytes behind the current output position to output;
Copy character 'f' to output.

利用此项技术被压缩的源数据文件，通过从开始到结束遍历已压缩的文件，可以去压缩而不需对前面读取的压缩数据追踪和重读。此外，虽然去压缩过程要求访问前面的去压缩数据，但去压缩数据也顺序输出。可以

- 15 用这种方式去压缩的压缩技术称为单调压缩/去压缩技术。

发明内容

本发明人知道，利用重叠匹配可获得进一步处理。重叠匹配是这样一种技术，欲编码的源文件数据的后面部分被包括在正在编码的重复部分中。例如，源串“abcabcabc”可以利用重叠技术进行如下形式的编码：

- 20 Copy characters 'a' and 'c' to output;

Copy a match of Size 6, starting 3 bytes behind the current output position to output.

- 25 换句话说，上面的字母串的开始为起点的 6 字母串与以第二个 'abc'（即在第一个 'abc' 编码后）为起点的 6 字母串匹配。利用重叠技术完成的压缩文件的长度（大小）要小于不利用重叠技术的文件长度。去压缩器用来处理编码的重复部分，方法是在某一时刻，把字符从原始单元重复部分复制到输出文件中的下一个单元。

- 30 为了缩小压缩文件的大小，现在的压缩技术是把指明要被直接复制的码字与包含有指向前面出现的内容保重匹配长度的返回指针的特殊码字进行优化组合，方法是改变这些码字的长度。一般来说，在要被直接复制的字符中，有些字符出现的次数比其他字符要多些。再者，对于典型数据来

说产生的返回指针和匹配长度的统计说明，给定长度的匹配数量在从 2 到约 6 或 7 之间几乎是指数下降，以后下降就不那么快。类似地，由于重复部分的参考区，有许多更小的返回指针（指明匹配模式接近于当前位置）。目前的压缩技术通过对特殊码字编码开发了这种返回指针和匹配长度的扭曲分布，并利用码字长度为连续可变的熵编码开发了未匹配的输入串。

本发明人知道，AVI 系统来说，对包含返回指针和匹配指针的特殊码字及复制字符码字并不适于采用真实的可变长度的二次压缩（熵编码），原因有二：第一，这种数据的去压缩将要求很贵的比特移位，因为码字不考虑字节边界。对于计算力较低的微处理器（例如，一次只能完成一个位置移位的 ST9）是一个瓶颈口。所以，应当把主要的精力用在去压缩路由部分上，以减少完成可变移位和转动所花费的时间。第二，真实的可变长度码字的快速去压缩要求在去压缩器中建立和保留查询表，而去压缩器必须花时间去建立和保留查询表，还须在系统 RAM 中分配空间去存贮这些查询表。因为在 AVI 解码器中仅有少量的系统 RAM（这就是为什么数据在第一位被送去压缩的理由之一），所以没有足够的系统 RAM 空间去建立足够大的查询表使能足够快地解码可变长度编码。

因此，本发明的主要目标是能快速去压缩，同时要求最小的系统 RAM 空间。根据本发明的一个方面，压缩的数据由受限可变长度码字表示。这种码字的长度受限，为多个半字节的码字，而不是真实的可变长度编码。这种方案只需要一定数量优化长度的码字，不要求大量的移位和转动操作，而这些都是真实的可变长度码字所要求的。

在目前的去压缩过程中，压缩的文件写入第一存储缓冲器。然后，使第二存贮缓冲器包含去压缩源文件。遍历第一存贮缓冲器中的压缩文件，把构成的源数据文件写入第二存贮缓冲器。这种技术要求分配两个存贮缓冲器，以便对压缩的文件去压缩。

发明人利用了单调压缩/去压缩技术，使压缩的文件就地去压缩。就是说，当收到压缩文件时，指定一个存贮容量足够的单缓冲器来容纳去压缩的源文件。压缩的文件就写入该缓冲器。然后，压缩的文件被去压缩成的单存贮缓冲器中的源文件。显然，这要求分立的缓冲器去容纳压缩的文件，使去压缩的存贮要求最小。

根据本发明的原理, 提供了一种对调节的单调压缩文件进行就地去压缩的方法, 所述压缩文件具有包含压缩码字的第一部分和包含简单正文的第二部分, 去压缩后的文件具有长度大小, 并包含简单正文, 其特征在于, 所述方法包括如下步骤: 指定一个具有去压缩文件长度大小的存贮缓冲器;

5 将所述的调节的单调压缩的文件存入所述缓冲器的底部; 和对调节的单调压缩文件第一部分中的压缩码字去压缩从而产生缓冲器顶部去压缩文件的相应部分的简单正文。

在上述方法中, 所述的去压缩步骤包括: 对缓冲器中某一位置设置一输入指针, 该位置相应于调节的单调压缩文件的起点; 对所述缓冲器的起点设置一输出指针; 从输入指针指向的所述缓冲器的所述位置读取码字并

10 前行输入指针; 解码所述码字去产生简单正文; 把所述简单正文写入所述输出指针指向的所述缓冲器中的所述位置, 并前行输出指针; 重复所述读取、解码和写入步骤, 直到所述输出指针相应于所述第二部分的所述位置。

其中, 所述调节的单调压缩文件还具有一定长度大小, 并包括表示该

15 调节的单调压缩文件长度大小、去压缩文件的长度大小和所述第二部分的位置的数据; 所述指定步骤包括读取报头的步骤以确定去压缩文件的长度大小; 所述存入步骤包括如下步骤: 读取所述报头以确定调节的单调压缩文件的长度大小和压缩文件的长度大小; 从调节的单调压缩文件的大小和去压缩文件的大小计算调节的单调压缩文件的起点在缓冲器中的位置; 和

20 把调节的单调压缩文件以计算的位置为起点存入缓冲器中; 所述去压缩步骤包括如下步骤: 读取所述报头以确定所述第二部分的所述位置; 对调节的单调压缩文件的起点在所述缓冲器中的计算位置设置一输入指针; 对所述缓冲器的起点设置一输出指针; 从所述缓冲器中输出指针指向的位置读取码字并使输入指针前行; 对所述码字解码以产生简单正文; 把所述简单

25 正文写入缓冲器中输出指针指向的位置, 并使输出指针前行; 重复所述读取、解码和写入步骤, 直到输出指针相应于所述第二部分的所述位置。

在所述第一部分中的所述压缩码字包括一代表了重复串中的字符位置和数目的码字, 所述重复串使用了重叠匹配; 和所述去压缩步骤包括对所述重复串去压缩的步骤, 所述重复串表示码字, 通过从重复串的去压缩文件的所述位置到去压缩文件的相应位置、一次一个地复制重复串中的字符

30

数完成去压缩过程。

所述压缩码字的长度是半字节的整数倍。

根据本发明还提供了一种把具有一定长度大小的源文件压缩为调节的单调压缩文件的方法，其特征在于，包括如下步骤：把源文件单调压缩为含有压缩码字的压缩文件；调节压缩文件以产生调节的单调压缩文件，该单调的压缩文件具有含有压缩码字的第一部分和含有简单正文的第二部分。

其中，所述调节步骤包括以下步骤：模拟指定一具有源文件长度大小的存贮缓冲器，把压缩文件模拟存贮到缓冲器的底部；设置一输入指针指向压缩文件的起点；模拟设置一输出指针指向缓冲器的起点；模拟去压缩压缩文件，该步骤如下：从压缩文件中读取输入指针指向的压缩码字，并使输入指针前行；模拟解码所述码字以产生简单正文；把所述简单正文模拟写入到所述缓冲器中输出指针指向的位置，并使输出指针前行；重复所述读取、模拟解码和模拟写入步骤，直到解码中模拟解码的码字使输出指针追上输入指针；和当模拟解码码字使输出指针追上输入指针时，相应于输出指针指向的源文件的位置从源文件中复制简单正文以产生调节的单调压缩文件。

优选地，所述方法中还包括把报头加到所述调节的单调压缩文件上的步骤，所述报头含有代表源文件大小、调节的单调压缩文件的大小和在含有简单正文的调节的单调压缩文件中的位置的数据。

其中，所述调节的单调压缩步骤使用了重叠匹配以产生压缩码字。

附图说明

图 1 是本发明使用的 AVI 信号解码器的一部分的框图；

图 2 是本发明一实施例，说明代表压缩文件中的源数数据的码字结构的存贮结构图；

图 3 是表示图 2 中报头 HDK 和码字 (CW1-CW8) 之间关系的更详细的存贮结构图；

图 4a-4j 示出代表了返回指针/匹配长度的各种码字格式；

图 5 和图 6 是了解压缩文件的调节使之就地去压缩的存贮图。

具体实施方式

图 1 是本发明所使用的 AVI 信号解码器的部分框图。该图所示的解码器被安装在每个 AVI 接收现场。在图 1 中, AVI 信号源(未示出)耦合到解码器的输入端 5。该输入端 5 耦合到 AVI 信号接收器 30 的输入端, AVI 信号接收器的输出端耦合到处理单元 40 的系统总线 416。处理单元 40 包括中心处理单元(CPU) 410, 例如是 SGS/Thomson ST9 微处理器; 读/写存贮器(RAM) 412 和只读存贮器(ROM) 414, 它们经系统总线 416 以公知的方式耦合在一起。

音频处理器 418 将音频信号提供给 AVI 音频输出端 25; 视频处理器 420 将视频信号提供给 AVI 视频输出端 15 和用户 I/O 适配器 424, 用户 I/O 适配器 424 经输入端 35 从用户接收数据, 所有这些装置都以公知的方法耦合到系统总线 416 其他装置, 例如连接计算机的 I/O 端口、调制解调器、计算处理器、其他 I/O 适配器等, 也可以公知的方式耦合到系统总线 416, 还可以包括一总线延伸器以便和那些与解码器外壳相连的其他盒状装置耦合。

在操作中, AVI 信号源, 例如, 直接的 RF 卫星链路、与解码器连接的有线馈线或光纤链路, 以包流的形式产生 AVI 信号。有些包载有音频分量, 有些包载有视频分量, 有些包载有相互作用分量。AVI 信号接收器 30 处理接收到的 AVI 信号, 提取音频、视频和相互作用分量, 并利用公知的 DMA 写入技术把他们包含的数据分别存贮到相应的预先规定的 RAM 412 中的存贮缓冲器中。然后, 音频处理器 418 和视频处理器 420 从 RAM 412 的各存贮缓冲器读取数据。然后, 音频处理器 418 和视频处理器 420 处理它们的数据分别在输出端 25 产生 AVI 音频信号, 在输出端 15 产生 AVI 视频信号。CPU 410 还可以在视频处理器 420 和音频处理器 418 的处理过程中与它们合作。相互作用包载有码和数据模块, 这些包在 CPU 410 的控制下以下面描述的方式被处理。

如上所述, 为了使相互作用分量的带宽最大, 应用程序设计员在把码和/或数据模块包括进 AVI 信号的相互作用分量之前, 可以决定利用单调压缩技术(例如 Lempel-Zip' 77 技术)压缩码和/或数据模块。此外, 重叠匹配可以用来进一步压缩模块。图 2 是根据本发明的优选实施例的一存贮配置图, 示出了在压缩的文件中代表了源数据的码字结构。在该优选实施例

中，压缩的文件是取连续的码字块的形式。每个块包括 1 位码-比特报头 HDR，其后是 8 码字 (CW1-CW8)。

图 3 是更详细的存贮配置图，示出了图 2 中报头 HDR 和码字 (CW1-CW8) 的关系。一个码-字节报头包含每个码字的相应比特标志，这些码字是连续的，在图 3 中，用箭头从报头的第一位到第一个码字以及从报头的第二位到第二个字节。如果在报头中的位是 1，则相应的码字是代表去压缩的数据字节的码-字节，去压缩的数据字节应当在去压缩过程中被复制成去压缩的文件。如果在报头中的位是 0，则相应的码字是返回指针/匹配长度码字。

图 4a-4j 是更详细的存贮配置图，示出了代表返回指针/匹配长度的各种码字格式。如上所述，码字在长度上是可变的，但受限为半字节 (4 位) 的倍数。在图 4 中，码字的长度 (即半字节的数目)、返回指针和匹配长度的值如下决定。如果第一位是 1，则码字或者是 2 或者是 3 个-半字节码字。如果第二位是 0，则匹配长度是 2。如果第 3 位是 0 (图 4a)，则它是 2 个-半字节码字，且后面的 5 位代表了返回指针 (BP) -1。如果第 3 位是 1 (图 4b)，则它是 3 个-半字节码字，且后面 5 位代表最低位 (LSM)，下一个-半字节代表了返回指针 (BP) -1 的最高位 (MSB)。

如果第二和第三位是 10，则匹配长度是 3。如果第 4 位是 0 (图 4c)，则它是 2 个-半字节码字，且后面的 4 位代表了返回指针 (BP) -1。如果第 4 位是 1 (图 4d)，则它是 3 个-半字节码字，且后面 4 位代表最低位 (LSB)，下一个半字节代表了返回指针 (BP) -1 的最高位。如果第二、第三、四位分别是 110，(图 4e)，则匹配长度是 4，码字长度是 3 个-半字节。后面 4 位代表了最低位 (LBS)，且下一个-半字节代表了返回指针 (BP) -1 的最高位 (MSB)。如果第二、第三和第四位分别是 111 (图 4f)，则匹配长度是 5 且码字长度是 3 个-半字节。后面 4 位代表最低位 (LBS)，且下一个-半个节是返回指针 (BP) -1 的最高位 (MSB)。

如果码字的第一位是 0，则匹配长度在该码字中被编码。对于这些码字，以如下方式决定匹配长度和返回指针。如果第二、第三和第四位分别不是 111 (图 4g)，则码字长度是两个码-字节 (4 个-半字节)。在这种情况下，第二、第三和第四位代表值 M。匹配长度是 M+3。剩下的 12 位 (4

位来自这个码-字节，8 位来自下一个码字节）代表值 B。返回指针是 B+1。如果第二、第三和第四位分别是 111，则码字至少是 3 个码-字节。如果后面的 5 位不是 11111（图 4h），则该码字由 3 个码-字节组成。第五、第六、第七、第八和第九位代表值 M。匹配长度是 M+3。返回指针（BP）是最后 5 15 位+1。这是返回指针的最大允许长度，如 32K。

如图第五、六、七、八和九位分别是 11111，则该字至少为 4 个码-字节长。对于这样的码字，返回指针最后面的 15 位+1 的值（与图 4h 相同）。如果最后的码-字节不是 255（如 11111111），（图 4i）则该最后的码-字节包含值 M。匹配长度是 M+34。然而，如果该字节包含值 255（如 10 11111111）（图 4j），则该码字实际是 6 个码-字节长。返回指针（BP）被包括在图 4C 的相同位置。最后两个码字节（字节 5 和 6）包含值 M。匹配长度是 M+289。

这个编码方法对于普通的返回指针/匹配长度使用了较小码字，只对较不常用的返回指针/匹配长度要求较大码字。此外，所有码字的长度都是半字节的倍数。通过把码字长度限制到半字节的倍数，所有的码字或者在存贮器的字节-边界开始，或者在存贮器中字节的半字节边界开始。因为很多 15 低-端微处理器，例如 ST9，具有调动一个字节内的两个-半字节的结构，所以能有效地提取码字，而不用进行太多的移位。况且，因为压缩器知道码字在存贮器中的位置，所以对它们编码的方法对去压缩处理过程来说， 20 检索就较快。

例如，如果码-字节 $X=b7.b6....b0$ 要放在输出字节 y 的半字节边界处，把它放在输出字节 y 和 $y+1$ 之间的方法应是：

$$y=x.x.x.b7.b6.b5.b4 \quad y+1=b3.b2.b1.b0.x.x.x.x \quad (1)$$

其中， x 是普通位（例如，它是 0 或 1）。提取码-字节 x 涉及从单元 y 和 25 $y+1$ 读取 16 位字 w ：

$$w=x.x.x.x.b7.b6.b5.b4.b3.b2.b1.b0.x.x.x.x \quad (2)$$

向左（右）进行 4 次移位操作。但如果压缩器已经对这个码-字节编码为：

$$y=x.x.x.x.b3.b2.b1.b0 \quad y+1=b7.b6.b5.b4.x.x.x.x \quad (3)$$

则该码-字节 能通过读取 16 位字 w 被从单元 Y 和 $Y+1$ 中提取出来：

$$30 \quad w=x.x.x.x.b3.b2.b1.b0.b7.b6.b5.b4.x.x.x.x \quad (4)$$

对 w 和分离字 M

$$M=0.0.0.1.1.1.1.1.1.1.0.0.0.0 \quad (5)$$

进行逻辑与，并对最后的 16 位字的两个字节相加，或逻辑或。在 ST9 上，通过这种优化节省的时间相当于先有技术方法中从 4 个移位操作中取消两个操作。

根据本发明，进行了如上压缩的压缩文件被调节从而允许就地去压缩。为了完成这个调节，原始源文件和压缩的文件的大小都要标出，且压缩文件进行第二次处理。

图 5 和图 6 是方便了解调节的存贮器图，进行调节是为了使压缩的文件能就地去压缩。在图 5 中，存贮缓冲器 100 画为竖直取向的矩形。缓冲器的总容量足以容纳去压缩的文件。为了进行对压缩文件的调节，通过试图就地进行去压缩，压缩过程连续地处理及先压缩的文件，并以如下方式监视其结果。压缩的文件 120 被写入缓冲器 100 的底部，如图 5 阴影矩形所示。设置一输入指针指向压缩文件 120 的第一位置，设置一输出指针指向存贮缓冲器 100 的第一位置，去压缩的文件将写入缓冲器。如以下所述，在压缩过程的调节相位期间，压缩文件 120 中的数据被从输入指针所指的位置读出，输入指针前进，写入存贮缓冲器 100 中输出指针所指的位置，输出指针前进，所有的方式都是已知的。

在压缩过程的调节相位期间，利用标准的单调去压缩过程去压缩，同时监视输入指针和输出指针的状态。图 6 是压缩处理的去压缩相位的过程，在图 6 中，包含数据的存贮缓冲器部分为阴影所示，不包含数据的部分是空白的（无阴影）。图 6a 相应于图 5，示出了在调节去压缩过程开始时的存贮缓冲器的状态，输入指针（IP）指向压缩文件 120（阴影）的起点，输出指针（OP）指向存贮缓冲器 100 的起点。

图 6b 示出去压缩过程刚刚开始后的存贮缓冲器 100 的状态。在图 6b 中，压缩文件 120 的第一部分已被读出，如输入指针 IP 向下读出所指示的。因为在去压缩过程中，单调压缩文件中没有回溯，所以存贮缓冲器 100 已经读出的部分将不会再次被读取，这部分适于容纳数据，如阴影所示。同时，响应于压缩文件 120 的已经读出的部分，去压缩文件 140 的第一部分已被产生。去压缩文件 140 的这个第一部分已被写入存贮缓冲器 100 的顶

部，如输出指针 OP 的移动所指出的。

图 6C 示出了去压缩继续进行的存贮缓冲器 100 的状态。如输入指针 IP 的位置所指示的，压缩文件 120 的大部分已被读出；如输出指针 OP 的位置所示的，去压缩源文件的大部分已被产生。从图 6C 可以看到，输出指
5 针 OP 的位置接近达到输入指针 IP 的位置。

如果输出指针 OP 追上输入指针 IP，就不可能继续解码压缩文件 120，因为去压缩源文件 140 的新写入部分将重新压缩文件 120 的将要读出的部分。图 6d 示出了这种情况。在图 6d 中，从压缩文件 120 读出下一个码字，如输入指针 IP 的移动所指出的。这个码字在图 6e 中以阴影矩形 CW 示出。
10 所示的码字 CW 是返回指针/匹配长度码字，它产生简单正文 PT，如图 6e 的阴影矩形所示。如所示，简单正文 PT 的容量长度为，当在输出指针 OP 位置写入缓冲器 110 时，它将重写要被读取的下一部分压缩文件 120。然后，输出指针 OP 将指向去压缩文件的新写入部分的底部，其在输入指针（IP）的值之下。这样，用输出指针（OP）结束了输入指针（IP），压缩
15 过程的相位调节监视着输入指针（IP）和输出指针。

根据本发明，压缩的文件 120 以如下方式调节。在压缩文件 120 中，从输出指针（OP）新的位置到压缩文件的结束的码字都被未压缩的源文件的简单正文所取代。最后的已调节的压缩文件 120' 包括了含有代表相应于未压缩源文件的码字的第一部分 124，和含有代表了相应于未压缩源文件的
20 的简单正文的数据的第二部分 126（如较重阴影部分所示）。已调压缩文件 120' 中的码字终止及简单正文开始的位置称为简单变址（PI），进一步标注。报头 HDR 被加到已调压缩文件 120' 的开始，它包含（于其他东西中）代表去压缩文件（DS）的长度大小的数据，压缩文件（CS）的长度大小及简单变址（PI）。

25 在一优选实施例中，上述的调节过程并不实际上从压缩文件 120 重新构成源文件 140，（如图 6 所示）。而只是模拟地重新构成。压缩文件 120 被遍历，且如果实际压缩被完成，输入指针（IP）和输出指针（OP）就被准确地保持在他们应当在的位置，但实际上根本没有产生去压缩文件 140。当发现在某一点需要在压缩文件 120 中进行调节时，就按如上方式用在第
30 一位置 124 的码字和第二位置 126 的简单正文重写压缩文件 120。然后把

报头加到调节文件 120' 上。

应当注意到，已调压缩文件 120' 的长度准确地与原始产生的压缩文件 120 相同（见图 6a-图 6d 的虚线所示）。也就是说，这个技术并不增加压缩文件的大小。

- 5 通过读取接收的调节压缩文件 120' 的报头 HDR，确定 RAM412 中（图 1）去压缩文件 140 的大小（DS），把没有报头 HDR 的调节的压缩文件 120' 存入缓冲器 100 的底部从而就地完成去压缩。调节的压缩文件 120' 的开始地址可以从压缩文件和无报头 HDR 的去压缩文件的长度计算出来。设定一输入指针（IP）指向调节的压缩文件 120' 的起点，设定一输出指针（OP）
- 10 指向缓冲器 100 的起点。然后，对调节的压缩文件 120' 的第一部分 124，采用输入指针（IP）和输出指针（OP），以类似图 6 的方式完成标准单调去压缩技术，直到输出指针（OP）的值等于简单变址（PI）。这样看来，只有简单正文被包括在调节的压缩文件 120' 中，不需要其他去压缩。如果利用两个缓冲器（即不是“就地”）完成调节的压缩文件 120' 的去压缩，
- 15 则当去压缩文件 140 中的输出指针（OP）达到简单变址（PI）的位置时，重设输入指针（IP）也指向调节的压缩文件 120' 简单变址（PI）位置，且调节的压缩文件 120' 的剩余部分被复制为去压缩的文件 140。

图 1

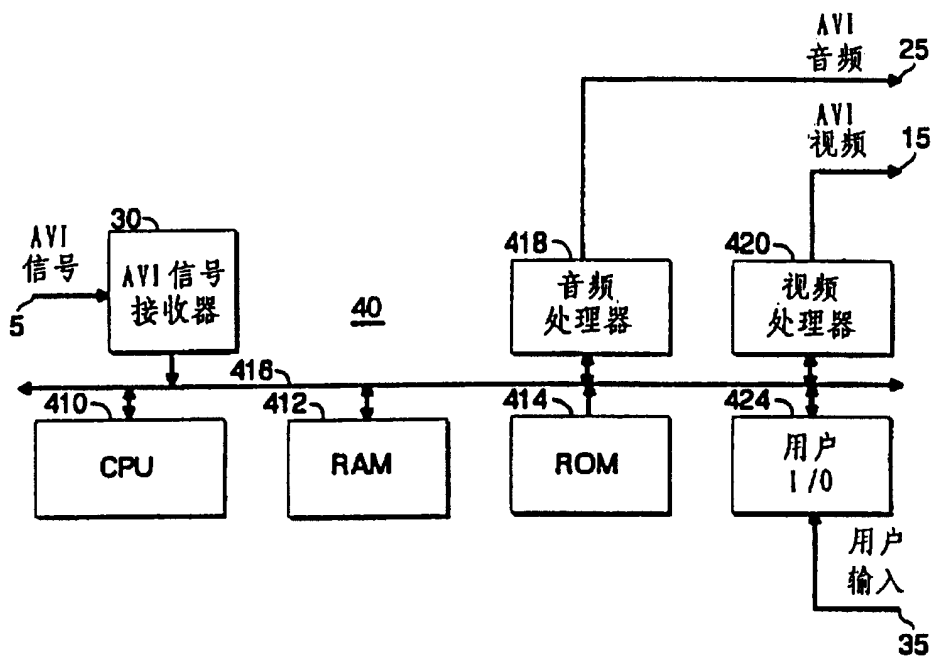
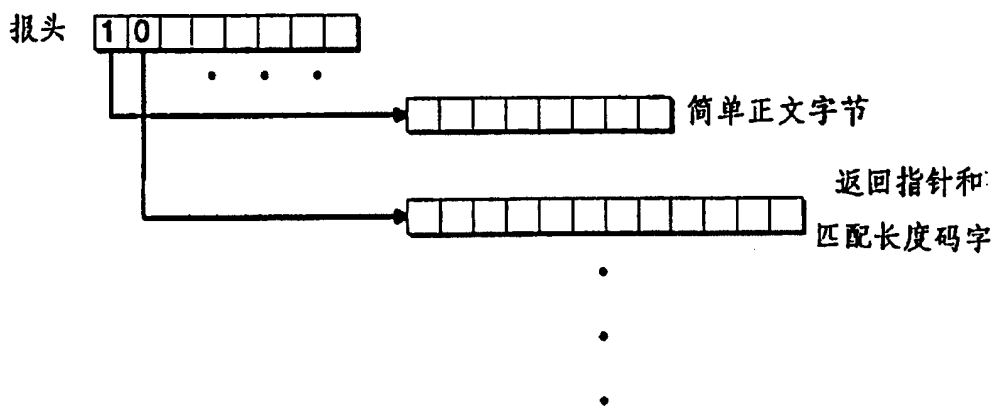


图 2

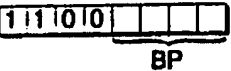


图 3



a)  MS = 2

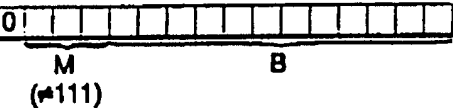
b)  MS = 2

c)  MS = 3

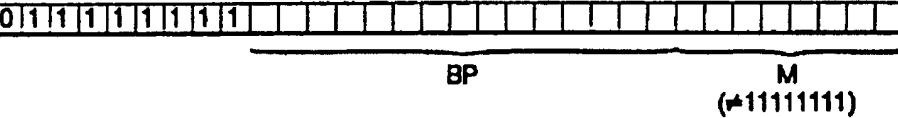
d)  MS = 3

e)  MS = 4

f)  MS = 5

g) 

h) 

i) 

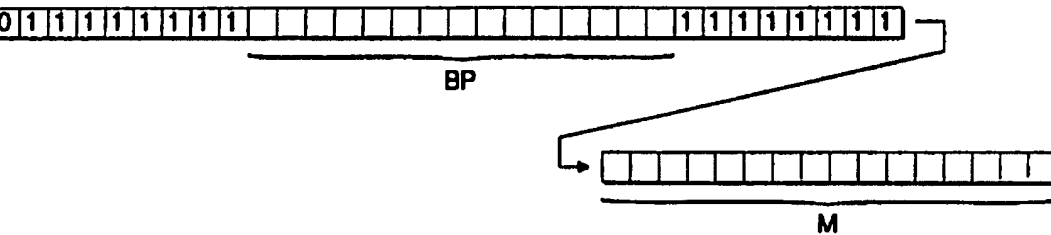
j) 

图 4

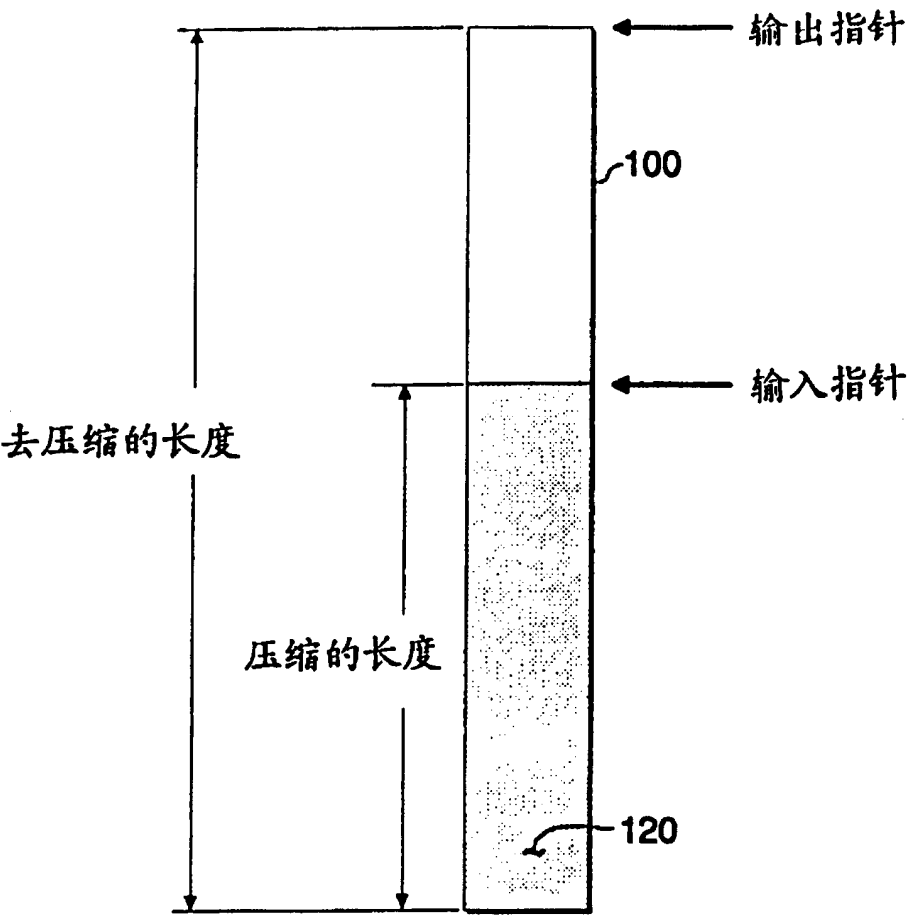


图 5

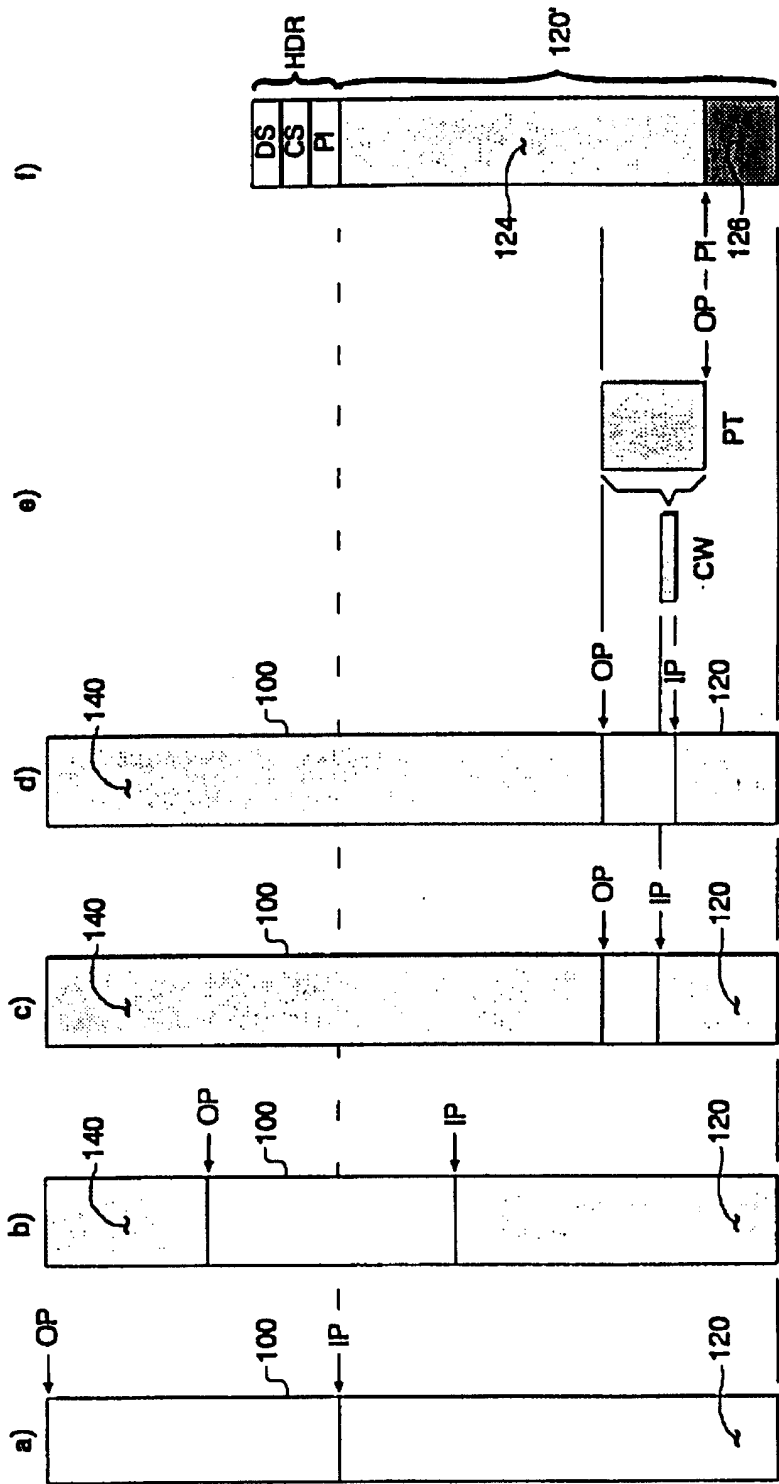


图 6