



[12] 发明专利说明书

[21] ZL 专利号 97181780.4

[45] 授权公告日 2004 年 6 月 9 日

[11] 授权公告号 CN 1153471C

[22] 申请日 1997. 12. 15 [21] 申请号 97181780.4

[30] 优先权

[32] 1996. 12. 18 [33] EP [31] 96402785.8

[86] 国际申请 PCT/US1997/022853 1997. 12. 15

[87] 国际公布 WO1998/027740 英 1998. 6. 25

[85] 进入国家阶段日期 1999. 8. 13

[71] 专利权人 汤姆森消费电子有限公司

地址 美国印安纳州

[72] 发明人 G·A·克拉纳维特

M·A·舒尔茨

审查员 李意平

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 王勇 陈景峻

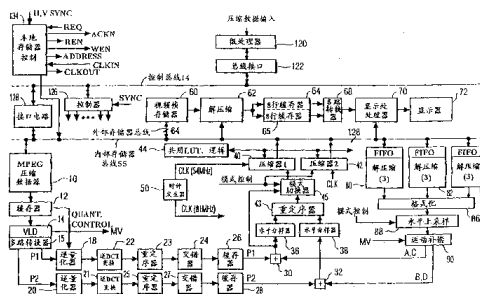
权利要求书 2 页 说明书 25 页 附图 21 页

[54] 发明名称 在一个 MPEG 译码器中用于再压缩的象素数据的并行解压缩器

[57] 摘要

在一个高清晰度的电视接收机中的一个 MPEG 译码器(14, 18, 22)译码和解压缩 MPEG 编码的数据以产生解压缩的图象象素块, 并且包括连接到一个帧存储器(60)的运动补偿网络(90...)以产生最终的译码的象素数据用于显示。在将解压缩的 MPEG 数据存储到帧存储器之前由多个并行的再压缩器(40, 42)对之进行再压缩。每个再压缩器接收一个交错的象素数据数据流(24, 27; 图 5)并且在每一时钟周期期间分别预测和压缩交错的象素值(a, c)(图 20, 27)。在再压缩之前对象素数据进行子采样的时候, 在一个数据缩减的处理模式中对一个再压缩器(42)撤去电源。在再压缩之前将子采样的数据进行重排序(43)。连接到帧存储器的多个并行解压缩器(80, 82, 84)将象素数据提供给运动处理网络。一个控制单元(356, 360, 364, 图 23)

在数据源被中断的时候通过重复最后一个有效数据确保将一个未被中断的交错的数据流提供给解压缩器。



- 5 1. 一个在用于处理一个运动图象专家组 MPEG 编码图象表示信息的数据流的系统中的 MPEG 兼容信号处理网络, 该信号处理网络包括:
- 一个存储器 (60), 用于存储从所述数据流导出的压缩的图象数据;
- 一个包括多个解压缩器 (80-84) 的控制环, 用于对存储于所述存储器中的所述数据进行解压缩并从所述解压缩的数据产生预测的图象数据; 以及
- 10 一个运动补偿网络 (90), 用于处理所述预测的图象数据, 以产生最终的解码的运动补偿的象素数据, 由一个显示处理器进行处理。
2. 根据权利要求 1 的网络, 其特征在于:
 所述数据流包含高清晰度视频信息。
- 15 3. 根据权利要求 1 的网络, 其特征在于:
 所述多个解压缩器位于所述存储器和所述运动补偿网络之间。
4. 根据权利要求 1 的网络, 其特征在于还包括:
 一个数字脉冲编码调制 DPCM 环, 其包括所述多个解压缩器, 所述存储器, 和所述运动补偿网络。
- 20 5. 根据权利要求 1 的网络, 其特征在于:
 所述解压缩器包括类似的并行解压缩器。
6. 根据权利要求 1 的网络, 其特征在于:
 所述存储器存储图象象素表示数据的压缩数据块; 以及
 所述多个解压缩器包括块解压缩器。
- 25 7. 一种处理运动图象专家组 MPEG 编码图象表示数据的数据流的方法, 包括步骤:
 存储 (60) 从所述数据流导出的压缩的图象数据;
 使用一个包括多个解压缩器 (80-84) 的控制环对存储在所述存储器中的所述数据进行解压缩, 以便从所述的解压缩的数据产生预测的图象数据; 以及
- 30 处理 (90) 所述预测的图象数据, 以产生最终的解码的运动补偿的象素数据, 由一个显示处理器进行处理。
8. 根据权利要求 7 的方法, 其特征在于:
 所述数据流包含表示高清晰度视频信息的数据。
- 35 9. 根据权利要求 7 的方法, 其特征在于:
 所述处理步骤包括运动补偿处理 (90)。

10. 根据权利要求7的方法，其特征在于：
存储步骤存储图象象素表示数据的压缩的数据块；以及
所述解压缩步骤使用并行的解压缩器（80-84）。
- 5 11. 一种在用于处理表示图象象素数据的压缩的MPEG编码信息的数据流的系统中用于产生适于由一个显示处理器处理的进一步的译码的象素数据的MPEG兼容译码处理方法，所述方法包括步骤：
解压缩（18，20，21，22）所述压缩的数据以产生第一解压缩的数据；
再压缩（40，42）所述解压缩的数据以产生再压缩的数据；
10 在存储器（60）中存储所述再压缩的数据；
使用多个解压缩器（80-84）对来自所述存储器的再压缩的数据进行解压缩以产生第二解压缩的数据；以及
运动补偿处理（90）所述第二解压缩的数据以产生所述进一步的译码的象素数据。
- 15 12. 根据权利要求11的方法，其特征在于：
所述数据流包括高清晰度视频信息。
13. 根据权利要求11的方法，其特征在于：
所述存储器是一个帧存储器。
- 20 14. 根据权利要求13的方法，其特征在于：
所述多个解压缩器包括用于处理前向象素预测信息的多个解压缩器和用于处理后向象素预测信息的多个解压缩器。

在一个 MPEG 译码器中用于再压缩的象素数据的并行解压缩器

5

技术领域

本发明涉及处理数字图象表示信息。

背景技术

10 数字技术的快速发展已在诸如高清晰度电视(HDTV)的各个领域的数字图象信号处理中产生了相应的进步。一个相关的发展是用于 MPEG-2 的视频处理的 MPEG(运动图象专家组)信号压缩标准(ISO/IEC 国际标准 13818-2, 1995 年 1 月 20 日)。已经发现这一广泛接受的图象处理标准在卫星, 有线电视, 和包括 HDTV 系统的地面广播系统的应用中特别具有吸引力。

15 在美国, 最近采纳为大联盟 HDTV 系统(Grand Alliance HDTV System)的数字 HDTV 地面广播系统定义了一种高清晰度(HD)节目素材的数字广播的标准, 该节目素材已经使用 MPEG-2 压缩标准进行了压缩。该大联盟 HDTV 系统的描述可在例如 1994 美国广播协会会刊的 1994 年 3 月 20-24 日第 48 届广播工程会议年会的会刊中找到(1994 Proceedings of the National
20 Association of Broadcasters, 48th Annual Broadcast Engineer Conference Proceedings, March 20-24, 1994)。该 HD 广播标准提供了高达每行 1920 个象素(水平)X1080 行的分辨率。该 MPEG-2 标准定义了用于解压缩该 HD 图象以由诸如在电视接收机中的显示设备用于再现所需的过程。如在地面广播标准中所定义的, MPEG 译码器正确地译码 HD 图像需
25 要大约 80 兆位(Mb)的存储器。在一个消费者接收机中通常需要 96Mb 的存储器。

在一个比如可以在电视信号接收机中发现的 MPEG 视频信号译码器, 为译码一个 MPEG 编码的数字数据流通常需要一个以上的图象帧存储器, 如所公知的, 这些帧代表 I, P, B 图象帧, 通常需要 3 个帧存储器以译码一
30 个 MPEG 数据流, 需要两个帧存储器来存储参考 I 或 P 帧数据和使用另外一个帧存储器存储 B 帧数据。

一个 MPEG 译码器包括一个 DPCM 环, 该 DPCM 环与用于产生最终译码的像素采样的运动补偿功能相关, 这是公知的。正如在共同未决的美国专利申请(序列号为 08/579192)中所公开的, DPCM 环被通过包括一个数字
35 压缩网络而进行有益的改动。这一压缩网络在将被解压缩 MPEG 数据传送

给帧处理器之前对该解压缩的MPEG数据进行再压缩,从而减少了对MPEG译码器的存储器需要。DPCM环使得要被压缩的像素值取决于预测器电路对要处理像素的紧邻的左边,正上方和对角线左上方的各个像素的计算结果。该预测器操作是一个实时的计算密集的串行操作。该预测器操作是重要的,因为涉及一个以上的像素值而且还因为好的压缩需要精确的预测而不是对像素值的“猜测”。

发明内容

根据本发明的原则,多个并行解压缩器从存储器向在一个MPEG兼容译码器中的运动补偿网络提供图象数据。

10 根据本发明的一个方面,提供一个在用于处理一个运动图象专家组MPEG编码图象表示信息的数据流的系统中的MPEG兼容信号处理网络,该信号处理网络包括:一个存储器,用于存储从所述数据流导出的压缩的图象数据;一个包括多个解压缩器的控制环,用于对存储于所述存储器中的所述数据进行解压缩并从所述解压缩的数据产生预测的图象数据;

15 以及一个运动补偿网络,用于处理所述预测的图象数据,以产生最终的解码的运动补偿的象素数据,由一个显示处理器进行处理。

根据本发明的另一方面,提供一种处理运动图象专家组MPEG编码图象表示数据的数据流的方法,包括步骤:存储从所述数据流导出的压缩的图象数据;使用一个包括多个解压缩器的控制环对存储在所述存储器中的

20 的所述数据进行解压缩,以便从所述的解压缩的数据产生预测的图象数据;以及处理所述预测的图象数据,以产生最终的解码的运动补偿的象素数据,由一个显示处理器进行处理。

根据本发明第三个方面,提供一种在用于处理表示图象象素数据的压缩的MPEG编码信息的数据流的系统中用于产生适于由一个显示处理器处理的进一步的译码的象素数据的MPEG兼容译码处理方法,所述方法包括

25 步骤:解压缩所述压缩的数据以产生第一解压缩的数据;再压缩所述解压缩的数据以产生再压缩的数据;在存储器中存储所述再压缩的数据;使用多个解压缩器对来自所述存储器的再压缩的数据进行解压缩以产生第二解压缩的数据;以及运动补偿处理所述第二解压缩的数据以产生所述

30 进一步的译码的象素数据。

附图说明

图1是根据本发明原则安排的利用一个MPEG译码器和相关的数字信号处理网络的一个电视信号接收机一部分的框图。

图2-17示出了有助于理解图1所表示的压缩/解压缩和相关网络的操作的像素块处理格式。

图18示出了像素的子采样(subsampling)和上采样(upsampling)。

图19是用于执行图18示出的过程的装置的框图。

图20示出了图1的压缩网络的细节。

图21示出了图1的解压缩网络的细节。

图22示出了有助于理解图20所示出的网络操作的像素布局。

图23-26示出了图1系统的数据流控制操作

图27是一个表示在图20示出的网络的操作期间像素关系的表。

图28示出了一个图23示出的网络的另一种安排。

具体实施方式

在本发明的所公开的一个实施例中，电视接收机中的 MPEG 译码器在该译码器和它的帧存储器之间利用了包括再压缩的数据缩减，从该译码器的帧存储器导出要显示的图像信息，预测器处理器的时序要求，本系统使用了流水线处理，其中必须得到3个像素（图像单元）的值，以预测一个给定的第4个像素的值，然而流水线处理降低了处理的速度（减少了带宽）。这个问题通过将来自 MPEG 解压缩器的独立的 8X8像素块的象素数据进行交错而解决。交错增加了处理的速度，因为它允许以交替的时钟处理数据，从而总是在产生压缩的像素值。该再压缩功能使用了数量缩减的压缩操作，用共享的功能进行交错的操作，从而节省了集成电路的面积。

图1示出了数字视频信号处理器的一部分，该信号处理器比如可以在处理一个输入的高清晰度的视频信号的电视接收机中看到。该视频处理器包括可在常规的 MPEG 译码器中看到的功能。例如在由Ang等人在题目为“视频压缩带来巨大的好处”（Video Compression Makes Big Gains）（发表于1991年10月IEEE Spectrum）的文章中描述了一个MPEG编码器和译码器。这些功能通常包括输入缓冲，可变长译码，逆量化和在产生最终译码的输出采样的相关的运动补偿处理之前的逆DTC变换。有关这些和相关的视频信号处理功能的其它信息，可以在Weiss的书“关于先进

的电视技术的问题” (Issues In Advanced Television Technology) (Focal Press, Boston, USA出版) 中找到。

图一的系统从单元10表示的数据源接收一个MPEG编码压缩的数据的控制的数据流, 该单元10包括一个在输入信号解调之后把数据包分开的
5 传送译码器, 在这个例子中所接收的输入数据流表示高清晰度图像素材 (1920像素/水平X1088个水平), 如在美国高清晰度地面电视广播系统的大联盟规范中所规定的。1920X1088高清晰度信息的数据速率是94003200字节/秒, 其由下式确定

$(1920HX1088VX30FX(8+4)YC)/B$ 其中:

10 H表示水平像素数

V表示垂直行数

F表示帧数/秒

YC表示(亮度+色度)位数

B表示8位/字节

15 在实践中通过内部存储器总线55和包括在单元128中的一个压缩的数据的接口提供所压缩的MPEG数据流, 单元128在微处理器120的控制下从控制总线114接收数据, 微处理器120通过压缩数据的输入端接收MPEG数据流。

来自源10的输入数据流采取数据块的形式, 每个数据块表示8X8个象
20 素, 这个数据表示压缩的编码的帧内信息和帧间信息。帧内信息包括I帧定位帧(anchor frame); 帧间信息包括预测的运动编码的残留图像信息, 该残留图像信息表示相邻图像帧之间的图像差别。该帧间运动编码涉及产生运动矢量, 该运动矢量代表在一个当前正在处理的数据块和在一个先前重构的图像块之间的偏移。将代表当前和先前块之间的最佳匹配的运动
25 矢量进行编码和传送。在进行传输之前也对每个运动补偿的8X8块和先前重构的块之间的差别(残留)进行DCT变换、量化和可变长度编码。这运动补偿编码过程的细节在各种公开物中都有描述, 其中包括上述的Weiss的教材和Ang等人的论文。

MPEG译码器要求的存储器容量减少了, 这就允许外部帧存储器的总
30 容量大大减小。正如随后将要解释的, 这是通过将要存储在存储器中的解压缩的视频帧进行再压缩和根据该译码器的操作模式通过选择性地水平过滤和分样(即子采样或下采样)在译码器环中的像素数据而完成的。比如在一中模式中该系统提供定位帧压缩, 在另一个模式中该系统通过低通滤波和下采样进行水平细节缩减之后提供压缩。

35 输入的压缩的像素数据块, 在被单元14进行可变长度译码之前由单

元12进行缓存,单元14如所知道的那样也产生运动矢量12。缓存器12在主层次(main level),主分布(main profile)MPEG数据流的情况下其存储器的容量是1.75兆位。通过一个多路转换器(Mux)15将译码的压缩的数据块从单元14输出,产生输出数据流P1和P2,输出数据流P1和P2表示双数据流水线,此后称之为流水线1(P1)和流水线2(P2),流水线P1包含一组用于一个给定的宏块的8X8像素块“A”的DCT系数,后跟一组用于该宏块的8X8像素块“C”的DCT系数。该DCT系数如所公知的被安排成对角线或之字形扫描格式。流水线1传送这种用于一个对应宏块序列的A,C块的序列,流水线2类似地包含一组用于该给定宏块和同时排成序列的多个宏块的DCT系数B和D,在流水线的序列中的这种像素块和宏块的像素数据的安排已被示出,并将结合图2-17讨论。

象素块数据由并行数据处理路径中的各个流水线进行传送,每个流水线包括一个逆量化器(18,20),一个逆离散余弦变换(DCT)单元(22,21),输出FIFO缓冲器(26,28),块重定序单元(23,25),块交错单元(24,27)和加法器(30,32)。在分别作用到加法器30和32的一个输入端之前,由该逆量化单元和逆DCT单元在各个流水线上分别进行解压缩和变换译码。

重定序单元23,25从单元21和22去除该逆DCT变换的象素数据之字形扫描图形,以对每个8X8块产生一个水平的逐行象素扫描序列。从而,例如在流水线1,单元23的输出表示形式为 $a_1, a_2, a_3 \dots a_{63} a_{64}$ (关于块A), $c_1, c_2, c_3 \dots c_{63}, c_{64}$ (关于块C)等的象素值。交错单元24使用多路转换技术产生形式为 $a_1 c_1 a_2 c_2 a_3 c_3 \dots a_{64} c_{64}$ 的流水线1的输出数据。对于块B,D,交错器27产生类似的序列。

逆量化器18和20的量化步幅由来自缓冲器的量化控制(Quant Control)信号控制,以确保有一个平滑的数据流。将所译码的运动矢量MV从译码器14提供给运动补偿单元90,正如下面将要讨论的。译码器14也产生一个帧内/帧间模式选择控制信号,这是众所周知的,为了简化附图,这里未予示出。单元14,18/20,21/22和23/25所执行的操作是MPEG译码器在发送器所执行的相应操作的逆操作。图1的MPEG译码器使用MPEG处理技术重建所接收的图象,下面将对MPEG处理技术进行简要描述。

基于视频帧存储器20的内容通过将来自单元26和28的残留图象数据与在运动补偿单元90的输出端所提供的所预测的图象数据进行相加而在加法器30和32的输出端分别提供重建的象素块。所重建的图象表示象素块的整个帧存储在帧存储器60中。在帧间模式中,使用来自译码器14的运动矢量MV从单元90提供所预测的块的位置。运动补偿处理

从以前所译码的图象中形成预测,该以前所译码的图象与(来自 IDCT 单元 21 和 22 的输出的)系数数据进行组合以恢复最后所译码的采样。运动补偿单元 90 按照例如在前面所提到的 MPEG 规范和 Weiss 和 Ang 的文献中所讨论的公知的 MPEG 兼容技术进行操作。单元 90 的 A, C 和 B, D 输出表示所解压缩的交错的象素块数据 A, C 和所交错的象素块数据 B, D, 如下面将对其进行讨论的。

图象重建过程涉及加法器 30, 32, 外部译码器帧存储器 60 和运动补偿单元 90, 由于使用了基于块的并行处理压缩器 40 和 42 和水平象素分样(子采样)单元 36 和 38(其减少了水平细节), 其优点是显著减少了对帧存储器的要求。作为由再压缩单元 40, 42 和单元 36 和 38 的分样所实现的数据缩减的函数, 帧存储器 60 的大小可以减少 25%, 50% 或更多。分样单元 36 和 38 的输出数据由块重定序单元 43 进行处理, 然后在利用水平分样的数据缩减操作模式中将输出数据传送到压缩器 40, 正如后面将要讨论的。结合图 12 和 14 以及相关的附图将可以看到重定序操作的效果。在激活水平分样单元 36 和 38 的缩减数据操作模式中, 模式控制 (Mode Control) 信号和模式切换器 45 修改压缩器的操作, 如将要讨论的。在缩减的数据模式中禁止压缩器 42(例如关闭电源)。在其他时间, 例如在处理高清晰度输入信号时, 将压缩器 40 和 42 都激活。

解压缩单元 80-84 执行由单元 40 和 42 执行的压缩操作的逆操作。单元 88 执行由单元 36 和 38 执行的分样操作的逆操作。后面将讨论这些操作的其他细节。格式化单元 86 摒弃不需要的解压缩的象素行, 直到获得包含单元 90 用于运动补偿预测器处理所需要的解压缩的象素的行时为止。这一选择过程防止累积不必要的的数据, 这一过程在存储器中不提供对每一象素的唯一映射的压缩系统(如所揭示的系统)中十分有用。关于这一方面, 请注意: 一个象素值可能已经以例如 3, 4 或 5 位进行了压缩或量化, 而只有在解压缩之后才知道该象素的值。

在再压缩之前通过交错 MPEG 译码的输入高清晰度数据流以及通过使用一个所交错的数据的压缩网络处理该所交错的数据进行该 MPEG 译码的输入高清晰度数据流的处理是有利的。该数据再压缩网络包括对一个所交错的数据流进行操作的类似的压缩器 40 和 42。这些压缩器共用包含在单元 44 中的某些逻辑电路和查找表, 以及响应由时钟发生器 50 产生的本地发生的 54MHz (2X27MHz) 时钟信号 CLK 进行工作。该 CLK 信号也作用到水平上采样网络 88。由发生器 50 产生的 81MHz (3X27MHz) 时钟信号也作用到解压缩器 62, 80, 82 和 84, 以及显示处理器 70。

在继续该系统操作的描述之前, 理解由图 2-17 所示出的交错的象素

数据处理操作的性质是有益的。图 2 示出了一个公知的 MPEG 宏块结构，该宏块结构包括亮度(Y)和色度(U,V)象素块部分。每一宏块的亮度部分由四个 8X8 象素亮度块 Ya, Yb, Yc 和 Yd 构成。色度部分包括所示出的 4X4 象素 “U”块 Ua-Ud 和 4X4 象素 “V”块 Va-Vd。交错器 24 和 27(图 1)如
 5 先前所讨论的和图 3 中的所示出的将数据流 P1 和 P2 中的这些象素块进行交错，图 3 示出如何对亮度和色度块进行安排以用于 A, C 和 B, D 流水线处理。在图 4 中相对于 4X4 象素块更详细地示出交错之前的流水线式处理，该 4X4 象素块构成一个 “U”色度部分。图 4 示出单元 23 和 25 将色度象素块 Ua 和 Uc 放入数据流 P1 和将象素块 Ub 和 Ud 放入流水线 P2
 10 的处理过程的结果。在该图中，A1 表示块 A 的第一个象素值(8 位)，A2 表示块 A 的第二个象素值(8 位)，B1 表示块 B 的第一个象素值(8 位)，如此等等，通过最后的值 A16 和 B16 至块 D。涉及亮度象素数据可以看到类似的处理。

在图 5-7 所示出的象素数据布局中，假定图 1 的单元 36 和 38 不进行
 15 水平分样(horizontal decimation)。在这一操作模式中，根据对于一个设定的系统配置所希望的数据缩减量(分样加上压缩)旁路 36 和 38。图 11-17 示出了允许水平分样的象素数据处理。

图 5 示出在流水线 1 中从交错器 24 的输出经过缓冲器 26 和加法器 30
 20 到压缩器 40 所传送的没有由单元 36 进行分样的交错象素数据的 A, C 序列。类似地，流水线 2 的象素数据以序列 B1, D1, B2, D2...等从交错器 27 和加法器 32 传送到压缩器 42。将宏块划分成由 A, B, C 和 D 数据组表示的段不是关键。比如，在另一个系统中，流水线 P1 可以传送 A, B 数据或 A, D 数据。类似地，流水线 2 可以传送一个数据组合而不是 B, D。在所示出的实施例
 25 中，流水线 1 所传送的 A, C 数据对应于 MPEG 规范中的“偶”数据块，流水线 2 所传送的 B, D 数据对应于 MPEG 规范中的“奇”数据块。

图 6 示出在 Huffman 编码后从第一个流水线的第一个压缩器 40 输出的所压缩的象素数据。图 6 数据流中的每一个 “x”表示一个“不关注”
 30 状态，该状态被产生以简化定时过程，从而对于每一时钟周期使用包含 8 位数据的连续时钟(而不是一个不太理想的停止/启动时钟)。写使能信号(未示出)确信当前只有有效的所压缩的数据被写到存储器。对于在输入端的每 16 个 8 位(色度)象素(8 个字节)，在输出端产生 16 个字节的所压缩的数据。对于流水线 2 的块 B, D, 没有示出从第二个压缩器所输出的类似的象素数据。将参考图 20 示出适合于压缩器 40 和 42 使用的压缩电路的细节，并进行讨论。

35 在单元 40 和 42 进行压缩之后，通过一个 128 位宽(即每一条传送一

位的 128 条并行数据线)的内部存储器总线 55(图 1)和一个 64 位宽的外部存储器总线 57 将该象素数据传送到外部译码器帧存储器 60。存储器 60 以解交错形式存储该象素块数据。可以在一个本地微处理器 120 的控制下,由与压缩器 40 和 42 相连的输出电路进行解交错,或者由存储器 60 5 之前的电路进行。这些电路使用已知的信号处理技术执行该逆交错功能,为了简化没有将这些电路示出在附图中。图 7 示出了在解交错之后传送到存储器 60 的压缩后的象素数据的形式。每个被压缩的象素用一个 3 到 6 位的数据表示。在所压缩的 A 数据块中,“a1”并不表示在该点的象素 a1,而是一个 8 位的数据,其由所压缩的象素和额外开销数据构成。一个象素的数据长度由该数据本身和该象素的位置确定。用于压缩在这一色度块中的数据的位数是 64 位。原始的色度数据由 128 位(8X16 位)构成。对“B”至“D”数据也进行类似的处理。

现在回头参见图 1,由包括显示解压缩器 62, FIFO 显示缓冲器 64, 多路转换器 68 和显示处理器 70 的显示处理网络对存储在存储器 60 的所 15 压缩的象素数据进行处理用于显示。显示缓冲器 64 容纳 16 个图象行,被分成两个分别为 8 行的缓冲器。从其中一个行缓冲器通过多路转换器 68 读出要进行显示处理的解压缩的数据,同时另一个行缓冲器以来自单元 62 的解压缩的数据进行填充。缓冲器 64 可以位于存储器单元 60。显示处理器 70 可以包括例如一个 NTSC 编码网络,用于调节显示象素的电 20 路,和用于向图象再现设备 72(例如高清晰度显象管或其他适当的显示装置)提供视频信号的显示驱动器网络。

在由单元 62 进行解压缩之前,对象素数据进行再次交错以提供一个“ab”块序列,如图 8 所展示的。这一交错可以由对存储器 60 寻址的适当的读出操作或通过显示解压缩器 62 相连的输入逻辑电路进行。类似 25 地,在解压缩之前,对象素 c 和 d 进行再次交错以产生一个“cd”数据序列(未示出)。该用于显示的再次交错序列(即 ab 和 cd)不同于最初的输入交错序列(即 ac 和 bd)。最初的交错允许例如首先对数据 a 和 b 进行访问以及对数据 a 和 b 进行并行处理。再次交错的显示序列适用于显示目的,其中需要来自同一图象帧的数据(象素 a, b, 和 c, d 在同一图象 30 帧)。图 9 示出用于“ab”序列的所交错的解压缩的象素数据序列。也产生用于“cd”序列的所交错的解压缩的象素数据序列(C1, D1, C2, D2, C3, D3 ...),但未在图中示出。在单元 64, 68 和 70 处理之后,将一个给定块的象素重新安排成如图 10 所示的显示格式。这是一个所简化的 4:2:0 形式而不是 4:2:2 形式的例子。

35 再次参见图 1, MPEG 译码器环也包括一个由与单元 86 和 88 分别进行

的数据格式化和水平上采样有关的多个解压缩器 80, 82 和 84 执行的解压缩功能。关于图 8 和 9 的上述描述也可应用于这一控制环解压缩功能, 其中在解压缩之前, 由与该解压缩有关的电路进行象素数据的交错, 以产生如图 8 所示的一个“ab”(和“cd”)数据序列。

5 图 11-17 示出了象素数据序列布局, 其中假设已经启动图 1 的单元 36 和 38 的水平分样(即子采样或下采样(downsampling))。在启动单元 36 和 38 的水平分样形式的数据缩减时, 由于数据量的减少, 禁止压缩器 42, 只使用压缩器 40 对数据进行压缩。网络 44 包含单元 40 和 42 所使用的逻辑电路和查找表。在数据处理需要量较少的数据缩减操作模式中, 在
10 一个压缩器不工作时, 只有另一个压缩器使用这些电路和表。在两个压缩器 40 和 42 都工作的高分辨率模式中, 利用该交错的数据流水线式的处理便利了对这些电路和表的共用。具体地, 单元 44 包含两个查找表, 一个由压缩器 40 使用, 一个由压缩器 42 使用。共用用于压缩器 40 的 LUT 来压缩所交错的 A 和 C 数据, 因为这些数据是在不同的时间被压缩的, 诸
15 如以交错的时钟, 后面将对其进行讨论。在压缩数据 B 和 D 期间, 类似地共用压缩器 42 的 LUT。

图 11 示出了从数据流 1 施加到图 1 的分样过滤器 36 的输入端的象素数据序列。过滤器 36 的分样产生图 12 的象素数据序列, 该序列也应用到重定序网络 43 的输入端。在图 12 中, 标记为“x”的单元表示“不
20 关注”或空数据, 在一个 H/2 模式, 用因数 2 对水平象素数据进行子采样, 过滤器 36 平均两个相邻的象素, 如公式:

$$a1^*=(A1+A2)/2,$$

$$c1^*=(C1+C2)/2,$$

25 $a2^*=(A3+A4)/2$, 等。这一过程示出在图 18 中, 如将要讨论的。也可以使用利用其他子采样因数的分样。图 13 和 14 类似地示出了图 1 中从数据流水线 2 施加到分样过滤器 38 的象素数据序列。

图 15 显示了在由图 1 的单元 43 进行分样和重定序之后的象素数据的序列。具体地, 由单元 43 对象素数据进行重新对齐, 将它们以适当的序列压缩和存储在存储器 60 中。在图 15 中, 象素数据 a1 至 c4 表示
30 在分样之后的一个图象帧(一个 2x4 矩阵), 象素数据 b1 至 d4 表示在分样之后的第二个图象帧(2x4 矩阵), 如此等等。

在图 1 中, 将来自单元 43 的所有重定序的象素数据通过模式切换器 45 作用到压缩器 40, 因为在经过分样之后, 只需要一个压缩器来压缩经过缩减的数据。在分样过程中的数据平均从两个象素产生一个象素, 产生更
35 少的数据, 并且相应的减少了对数据处理带宽的需要。因此只要一个压

缩器就够了, 压缩器 42 处于不工作状态。如以下所讨论的, 压缩器响应模式控制信号根据分样是否操作对压缩器进行使能和禁止。

5 发生在重定序网络 43 中的重定序不象先进先出缓冲器中那样是一个直接的过程。为了减少包括水平分样, 重定序和再压缩的运动补偿环处理的复杂性, 以基本上与没有经过单元 36 和 38 进行分样处理的数据相同的格式将数据提供给压缩器 40。提供独立于再压缩器 40 的重定序网络 43 简化了电路, 因为重定序器 40 不必区分来自单元 36 和 38 的要求重定序的数据和来自加法器 30 和 32 的不要求重定序的数据。

10 图 15A 是图 11 至 15 的汇集, 显示了相对于像素时序经过水平分样和重定序的数据流。图 15A 的数据流 15-1 和 15-2 分别表示从加法器 30 和 32 出来的和进入分样器网络 36 和 38 的数据。数据流 15-3 和 15-4 分别表示从分样器网络 36, 38 出来并且进入重定序网络 43 的数据。如前面所讨论的, 对像素进行交错, 如在数据流 15-1 至 15-4 中所看到的。数据流 15-1 和 15-3 表示来自流水线 P1 的数据, 数据流 15-2 和 15-4 15 表示来自流水线 P2 的数据。数据流 15-5 表示从重定序网络 43 出来的数据, 该数据经过切换器 45 进入再压缩器 40。在图 15A 的底部, 是像素时钟 CLK, 以表示数据像素通过该系统时的时序。作为一个示例, 将所选择的数据像素通过重定序过程。该过程对任一流水线的的数据是相同的。该数据流 15-1 和 15-2 的像素表示一个色度像素宏块。对于亮度像素宏 20 块该过程是相同的, 但是该过程更复杂, 因为其再现是在四个 8x8 的像素块上而不是四个 4x4 的像素块上扩展的。该较大的宏块使得重定序在更多数目的时钟周期以四倍的数据量进行。然而, 该重定序的原则对于亮度和色度数据是相同的。

来自数据流 15-2 的像素 B1 被分样至百分之五十, 并且与像素 B2 组合 25 以形成与一个输入数据像素同样大小的一个输出数据像素。对于像素 D1 和 D2 也是同样的。分样器网络 38 缓冲来自像素 B1 和 D1 的被分样的数据, 直到像素 B2 和 D2 被处理。这就是在第一个两个时钟周期期间来自分样网络 38 的输出数据无效的原因。在第三个时钟周期期间出现有效数据, 如像素 b1*。在第三个时钟周期的前半部分期间输出像素 B1 的数据, 在第三个时钟周期的后半部分期间输出像素 B2 的数据。在第四个时钟周期以 30 同样方式产生像素 d1*。

将来自流水线 P1 和 P2 的数据传送到重定序网络 43, 该重定序网络 43 缓冲该数据并以适当的顺序访问特定的像素, 以形成进入压缩器 40 的一个连续数据流。如同从图 15A 的数据流 15-4 和 15-5 所看到的, 像素 35 b1*, b2*, b3* 和 b4* 必须与像素 d1*, d2*, d3* 和 d4* 相交错, 但是在相应的 a, c

象素之后。因此这些象素驻留在重定序网络43中不同的时间，以等待进行输出。比如在时钟周期3期间重定序网络接受象素b1*，并且在时钟周期12期间输出，而在时钟周期7期间重定序网络接收象素b2*并且在时钟周期14期间进行输出。通过由一个微处理器120所控制的状态机将象素

5 5 导引到重定序网络43。

为了维护恒定的数据流，压缩器40期待以交错格式的输入象素数据，如在数据流15-1和15-2中所示出的。在分样网络36和38之后改变象素的顺序，因为两个流水线P1和P2被按照2的级数下采样，每一个流水线提供数据流15的一半数据。然而来自P1和P2的下采样的数据源于该图象的

10 10 垂直相邻的块。压缩器40等待来自水平相邻块的所交错的象素数据。因此重定序网络43将以数据流15-3和15-4所示出的顺序的下采样的数据组合成数据流15-5中所示出的顺序。这一顺序实质上与没有经历过在分样网络中的下采样的所交错的的数据相同。来自下采样的数据和没有经过下

15 15 采样的数据的象素块的大小是相同的，也就是它们水平和垂直方向具有同样数目的象素。唯一的差别是所下采样的象素数据块包括来自两个水平相邻象素块的象素信息，如前面所描述的。这个差别对于压缩器40是透明，该压缩器允许连续的数据流。然而这一系统进行重定序以将水平相邻的象素块组合成一个下采样的象素块，本发明的精神也包括这样一个系统，该系统组合具有不同的空间关系的象素块。

20 20 如图15A所看到的，重定序网络43表现为需要来自分样器网络36的象素a2*至a4*和a6*至a8*(数据流15-3)以在它们可用之前输出(数据流15-5)。现实是，这不能发生也不会发生，但其所示出的是重定序网络43必须适应的不同的时序和延迟。为了防止数据在被重定序网络43接收之前而需要输出，单元43保存并且延迟足够数量的数据，直到可以处理所有的

25 25 数据，因此提供了如在数据流15-5所示出的连续的数据输出。这一延迟对于流经流水线P1和P2并且到达重定序网络43的第一数据发生，比如在对电视接收机开始加电时，在频道切换时，或者在任何时候建立数据同步时发生这种延迟。在一个最初的延迟之后，数据就是连续的了，不损失时钟周期。

30 30 图16描述了从压缩器40输出的所压缩的象素数据的序列。在图16中，所压缩的数据“m”指示由从分样后的象素a和b(即对一个8X8象素块进行分样后所产生的一个4X8象素块)导出的数据构成的所压缩的象素数据。类似地，所压缩的数据“n”指示由从分样后的象素c和d导出的数据构成的所压缩的数据。象素数据a和b在同一个图象场，象素

35 35 数据c和d在同一个图象场。由压缩器40进行的象素块压缩过程被设

计成对 8X8 象素块进行操作。在分样之后,将所产生的 4X8 象素“a”块和 4X8 象素“b”块进行组合以产生一个 8X8 象素块,其被压缩以产生象素块“m”。类似的作法涉及从所分样的 4X8 象素块“c”和“d”形成所压缩的象素块“n”。在这一方式中,为进行有效的 MPEG 译码,将在同一个
5 图象帧中的象素块进行适当的对齐。图 17 表示了被传送到并且存储在存储器 60 中的正确的帧序列的所压缩的象素块的安排。

通过减少存储在存储器 60 中的象素值的数量,由分样网络产生的水平细节缩减还减少了对译码器的存储器要求。在提供数据给存储器 60 之前,分样网络 36, 38 利用一个水平空间低通滤波器,后跟一个 2:1 水平
10 分样(下采样)。在由单元 80, 82 和 84 进行解压缩之后,来自存储器 60 的图象信息的分辨率由单元 88 使用象素重复上采样过程重新构成。该上采样过程在显示解压缩器 62 和显示处理器 70 之间不需要,因为处理器 70 提供了所要求的水平采样率转换。在缩减成本的接收机中,预计显示解压缩器 62 和处理器 70 将不执行上采样,因为这样的接收机提供的
15 显示分辨率减少了。在这一情况下,存储器缩减了的所译码的帧的分辨率高于标准清晰度的显示器。比如,为了译码和在一个 720X480 象素显示设备上显示一个 1920X1088 象素的视频序列,就要求存储在帧存储器中的图象具有 960X1088 的分辨率(进行 1/2 水平分样)。从而显示解压缩器 62 不必需要上采样图象,但是显示处理器 70 将不得不对 960X1088
20 分辨率的图象进行下采样成 720X480,以适于在显示器上显示。

图 18 和 19 分别示出了与由图 1 的单元 36, 38 执行的象素子采样过程和由单元 88 执行的象素上采样有关的单元的总的布局。在单元 36 和 38 中,在进行二分样之前,该原始的象素首先由一个偶序号低通滤波器
102 进行低通滤波,从而单元 104 每隔一个去除一个象素值。这些象素值
25 存储在存储器 60 中。以后,由上采样单元 88 的部件 106 使用公知的技术重复存储器 60 中的象素数据。

在这一例子中,过滤器 102 是一个 8 抽头对称 FIR 过滤器。这一过滤器在该水平空间域中操作,并且跨越块边界进行过滤。该 8 抽头过滤器的作用是将输出象素的相对位置相对于该输入移动半个采样周期,如图
30 18 所示。正如在图 18 所示出的,象素重复上采样的效果是保持该所下采样/上采样的象素相对于该原始象素的同一空间位置。分样过滤器单元 104 可以是两抽头过滤器,使得对于输入象素 x 和 y ,该过滤器输出是 $(x+y)/2$,该分样通过每隔一个象素去掉一个而完成。该过滤器并不跨越块边界,容易实现,并且是水平分样的较好选择。

35 图 1 所示的电视接收机系统已被简化,使得不会用过多的细节使附图

复杂。例如,没有示出与该系统的各个部件有关的 FIFO 输入和输出缓冲器,读/写控制,时钟发生器电路和用于连接到外部存储器的控制信号,该存储器可以是扩展数据输出类型 (EDO) 的存储器或同步型 (SDRAM) 的存储器。另外图 1 的系统包括一个用于发送和接收数据,例如读/写使能和地址信息的微处理器 120, 5 5 连接到内部控制总线 114 的总线接口 122 和控制器 126。在这一例子中,微处理器位于包括该 MPEG 译码器的集成电路的外部。

显示处理器 70 包括将一种所译码的图形格式转换成单元 72 用于显示的一种预定的格式所需要的水平和垂直再采样过滤器。例如,该系统 10 可以接受和译码对应于诸如 525 隔行, 1125 隔行或 720 逐行扫描的格式的图象序列。处理器 70 也产生与该图象显示器有关的时钟和 H, V, 同步信号, 以及通过内部存储器总线 55 与帧存储器 60 通讯。

外部总线接口网络 122 除了输入所压缩的视频数据用于由该 MPEG 译码器处理之外, 还在 MPEG 译码器和外部处理器 120 之间传送控制和配置信息。该 MPEG 译码器系统模仿微处理器 120 的协处理器, 例如对于 15 要译码的每一帧, 微处理器发布一个译码命令给 MPEG 译码器。该译码器定位该相关的头标信息, 然后该头标信息由微处理器 120 读取。利用该信息, 微处理器 120 发布用于配置该译码器的数据, 即关于帧类型, 量化矩阵等, 在这之后, 译码器发布适当的译码命令。可变长度译码器 14 20 通过存储器总线 55 进行通讯, 并且接口电路 128 提供了存储器总线 55 和控制总线 114 之间的通讯。

模式控制数据由接收机的制造厂家进行编程, 该数据由与存储器控制器 134 和控制器 126 相连的微处理器 120 进行传送, 如按系统的设计参数所要求的以建立用于单元 40, 42 和 80-84 的压缩/解压缩因子和控制 25 压缩和解压缩网络以及上采样和下采样网络的状态。微处理器 120 也将存储器 60 分成多个帧存储器段, 帧存储器和位缓冲器, 以及用于 MPEG 译码和显示处理的在屏显示位映射段。本地存储器控制单元 134 接收(例如来自单元 70 的)水平和垂直同步, 和数据请求输入, 以及提供确认输出和存储器地址, 读取使能 (Ren) 和写入使能 (Wen) 输出到包括缓冲器控制 30 电路的各种系统电路。单元 134 为控制存储器 60 产生实时地址和控制信号。响应信号 CLKin 中的输入时钟, 提供输出时钟信号 CLKout, 时钟信号 CLKin 可以由诸如单元 50 的本地时钟发生器进行提供。图 1 的系统可以与在各种数字数据处理方案的上下文中的 MPEG 规范的所有层面 (Profile) 和级别一起使用, 这些数据处理方案可以与例如地面广播, 有线 35 电视和卫星传送系统有关。

在这一实施例中,视频帧存储器 60 位于包括图 1 的 MPEG 译码器和有
关部件的集成电路的外部。显示处理器 70 可包括一些在 MPEG 译码器集
成电路上没有的部件。使用这种外部存储器器件允许接收机的制造商选
5 择经济的存储器器件,使得在接收机接收到高清晰度的数据流时,能与接
收机的预定使用目的,例如全高清晰度显示器或清晰度降低的显示器,相
兼容。作为实际应用,通常用于 MPEG 译码的较大容量的存储器当前要求
该存储器位于该译码器集成电路的外部。技术的未来发展有可能允许该
存储器与该 MPEG 译码器部件位于同一个集成电路上。然而使用外部存
10 储器器件给了制造商选择存储器的大小的自由,以与接收机的显示器分
辨率和其他特征一致。

在实际上,接收机制造商将决定是将接收机配置为一个具有扩展特征
的昂贵的优质模型,还是配置为具有较少特征的更为经济的模型。一个
感兴趣的特征是所显示的图象的分辨率。在一个成本节约的接收机中,
有助于节约成本的因素包括廉价的分辨率降低的图象显示设备和与 MPEG
15 译码器有关的存储器的容量。

在这一例子中,在压缩器网络将数据压缩 25%时,对存储器的要求降
低到 64 兆位,并且在压缩器网络将数据压缩 50%时,对存储器的要求降
低到更为经济的 48 兆位。该 25%的压缩因子与全 HD 图象显示器有关,并
且几乎与没有压缩的全 MPEG 译码没有差别。采用 50%的压缩比例,经过
20 训练的观看者能够发现几乎注意不到的人为现象。在任一情况下,所译
码的图象序列提供全 1920X1088 的 HD 分辨率,用于由一个全 HD 分辨率
图象显示装置显示。

在有些情况下,不需要全 HD 图象分辨率,诸如在一个型号的接收机使
用的廉价的显示装置的分辨率低于全 HD 分辨率时。在这种情况下,最好
25 接收和译码 HD 信息而不显示全 HD 分辨率图象。在这样一个接收机中,
分样器网络 36, 38 和压缩器网络 40 可以被一起用于极大地减少译码器
的存储器要求。例如,分样器网络可以以 1/2 水平分样数据,压缩器网络
可以将所分样的数据压缩 50%。这就会将译码器的存储器的容量极大地
减少,只有 32 兆位。在这一情况下,用于显示的图象的分辨率是 960X1088,
30 这对于 1H 或 2H 的接收机应用是足够的。从而只使用 32 兆位的 MPEG 译
码器存储器就可以构成能够译码全 HD 图象数据流的低成本接收机。
响应由微处理器 120 提供给切换器 45 的模式控制信号,执行上述的操
作。根据该 MPEG 译码器是位于一个高清晰度的接收机中还是位于一个
分辨率低的接收机中,编程微处理器 120 以确定压缩的量,以及确定是否
35 启动分样器网络对数据下采样,或对其旁路。

图 1 的系统展示了用于处理一种信号的第一数据处理模式和用于处理一种数量缩减的数据的第二模式,该信号包含诸如由高清晰度显示装置显示的一种高清晰度电视信号的大量的数据。该第二模式例如可以实现在一个包括具有较低的数据分辨率功能的成本较低的显示装置的经济
5 的接收机(即一种非 HDTV 显示装置)中。

图 1 中切换器 45 的状态由模式控制信号控制,该信号可由接收机制造商进行编程以指示由该接收机显示的数据的类型,例如高清晰度(第一模式)或低于高清晰度(第二模式)。如果所接收的高清晰度信号要由高清晰度显示装置显示,则控制切换器 45 以产生第一模式操作,而在一种
10 高清晰度输入信号由单元 36 和 38 进行子采样以产生低于高清晰度的图象信息用于由具有低于高清晰度分辨率的经济的显示设备进行再现时,控制该切换器产生第二模式。

在第一模式,分样器单元 36 和 38 被旁路,要压缩的数据块被直接传送到压缩器 42 和通过切换器 45 传送到压缩器 40。在这一模式中,将模式控制信号作用到压缩器 42 的控制输入端,以启动压缩器 42。在第二模式
15 中,在这一实施例中,通过移去压缩器 42 的电源,该模式控制信号的状态禁止压缩器 42,同时通过切换器 45 使能要从加法器传送的数据以激活加法器 40。在一个用于处理 HDTV 信息的集成电路器件中通过移去电源而禁止压缩器 42 是极其有利的,这是因为由于这种集成电路的高的时钟
20 频率,大的表面积和集成在其上面的工作部件的数量很大,对其功耗(散热)有限制。在一个选通的时钟系统中,移去电源可通过停止压缩器的时钟来有效实现。这种操作的一个另外的优点是压缩器只需要以类似于诸如 8X8 和 4X8 的块处理模式的方式操作。也就是,例如不需要对压缩器 40 进行再编程以处理由该分样处理产生的 4X8 象素块。块重定序单元 43
25 从 4X8 象素块重建分样后的象素块以产生与该压缩器的块处理算法兼容的 8X8 象素块。

在没有利用单元 36 和 38 进行分样时,也将模式控制信号作用到水平上采样网络 88 的控制输入端,以旁路在操作模式中的上采样功能。为此目的,单元 88 可以利用一个相对简单的切换装置,以在这种旁路模式中
30 将单元 86 的输出信号直接切换到单元 90。

在存储数据到存储器 60 之前的压缩要求在运动补偿处理环中的单元 88 之前对数据进行解压缩。这是通过基于块的解压缩器 80,82 和 84 完成的,解压缩器执行压缩器 40 和 42 的逆操作。基于块的显示解压缩器 62 使用类似于压缩器 80-84 的解压缩技术,在将所存储的象素数据传送到显示处理器 70 之前对之解压缩。在存储器 60 之前启动下采样网络
35

36, 38 时, 在运动补偿处理环中的单元 90 之前由单元 88 对存储器 60 的输出数据进行上采样, 该操作是网络 36 和 38 的逆操作。

有利的是, 图 1 的系统利用多个并行块解压缩器, 它们分别用运动补偿环中的单元 80, 82 和 84 表示。在这一例子中, 使用了 9 个解压缩器, 在单元 80, 82 和 84 中每一个有 3 个解压缩器, 以允许所有的象素单个被解压缩。这些解压缩器的每一个具有一个相关的 FIFO 输入缓冲器。(例如在单元 80 的) 三个解压缩器被用于解压缩在一个 MPEG 前进预测模式中的亮度信号, (例如在单元 82 的) 三个解压缩器被用于解压缩在一个 MPEG 后退预测模式中的亮度信号。由于色度信息是亮度信息的一半, 仅仅 (例如在单元 84 中的) 的三个解压缩器被用于解压缩色度象素数据。对于最坏情况下的 MPEG B-图形译码, 需要使用所有 9 个解压缩器, 这时需要双向运动补偿预测处理。从而 B 图形预测要求两个图象帧 (前一帧和最后一帧), 同时 MPEG P 图形预测只需要一个图象帧。

可能 (并且经常不会) 在一个块边界不出现运动补偿预测器块 (predictor block)。相反, 几个块可能不得不从帧存储器 60 中调用。在最坏的情形下, 在一个具有一半象素分辨率的 MPEG-2 系统中, 该运动补偿预测器块可能交叉 6 个块。从而必须从存储器访问 6 个块。在一个诸如具有在运动补偿环中 (通过单元 40, 42) 进行再压缩的所揭示的系统中, 不能直接访问象素。所有的块象素必须首先进行解压缩, 在出现 6 个块的最坏的情形下这需要许多开销, 并且产生的数据比实际需要的多。如前面所描述的, 格式化单元 86 将不需要的象素信息丢弃, 但是仅仅在所有的象素已被解压缩之后。

在要处理大量数据的情形下, 诸如在如上所述的 6 个块的情形下, 在存储之前进行解压缩极大地增加了与处理该所解压缩的象素信息有关的缓冲器容量的要求。相反, 已经发现最好在所揭示的系统中如所揭示的那样并行地解压缩数据, 并且在后来 (通过单元 86) 丢弃与该预测器块不相关的不需要的所解压缩的象素数据。有利的是, 这一过程要求明显小的缓冲器存储容量。因此, 尽管降低了缓冲器存储器带宽 (数据容量) 的要求, 但需要更多的集成电路表面积。然而, 并行地使用几个解压缩器带来的其他优点是, 对运动补偿预测器处理所需要的数据的操作更快, 以及随之带来的对之快速的访问。

该多个解压缩器不是成流水线。每个解压缩器和其相关的缓冲器独立地分发数据, 从而快速地分发数据。在一个解压缩器/缓冲器网络中操作的延迟不影响其他解压缩器网络的操作。解压缩器也对于象素数据执行交错的操作, 这便利了每个解压缩器的独立操作。也象该压缩器网络,

解压缩器 80, 82 和 84 共用单元 44 中的一个公共的查找表 (LUT)。

根据特定系统的要求, 网络 40, 42 可以使用包括量化和变换的各种压缩类型。所揭示的系统使用固定长度压缩, 尽管也可以使用可变长度压缩或自适应固定/可变压缩。

- 5 所使用的压缩的类型最好具有某些特征。每个块应当被压缩一个预定的量, 以使得容易确定每一所压缩的块的位置。每一块应当独立于其他块压缩/解压缩。从而不需要读取任一其他块就可访问任一块。该压缩/解压缩过程不应当在所再现的图象中产生可以看到的人为现象。与没有这种压缩的常规的译码器处理相比, 以 25% 的压缩比几乎是清楚的。
- 10 以 50% 的压缩, 其结果是清楚程度降低, 但其结果是可接收的, 与没有压缩和存储器缩减的常规译码器相比, 不认为是明显地不同。

图 20 示出在图 1 的每一再压缩器 40 和 42 中使用的固定的压缩网络。该压缩网络利用了具有自适应预测的差分脉冲编码调制 (DPCM) 环。这种具有预测的 DPCM 处理的思想是消除连续象素之间的相互冗余, 以及只产生差别信息。这一公知的过程在 A. K. Jain 所著的“数字图象处理基础” (Prentice Hall 国际出版社出版) 第 483 页中有描述。

15

在讨论图 20 的电路之前, 现在参照图 22 进行描述。图 22 示出了与 DPCM 网络的预测处理操作相关的一个四象素组 a, b, c 和 x (要预测的象素) 的一种示例安排。在图 22 的 8X8 象素块中引用该象素组。以图 22 所示出的光栅方式扫描每一象素块, 方向是从左到右, 从上到下。在这一例子中, 对于亮度信息, 相对于象素 c 延迟象素 b 一个象素间隔, 相对于象素 b 延迟象素 a 七个象素间隔, 相对于象素 a 延迟象素 x 一个象素间隔。对于色度信息, 象素 “a” 延迟三个象素间隔。

20

在 DPCM 预测处理中, 通过使用先前编码的象素预测要编码的当前象素, 对于解压缩器 62, 80, 82 和 84 (图 1) 这是已知的。在图 22 中, 象素 x 是要预测编码的象素值, 象素 a, b, 和 c 先前已经进行了预测编码, 这对于解压缩网络是已知的。根据下面的伪码, x 的预测 Xpred 使用 a, b, 和 c 的值, 该伪码描述了所使用的算法逻辑:

25

```

if          (|a-c|<e1 && |b-c|>e2), Xpred=b
30  else if   (|b-c|<e1 && |a-c|>e2), Xpred=a
   else                                           Xpred=(a+b)/2

```

其中 e1 和 e2 是代表预定阈值的常数。只对于没有位于该要处理的块的第一行或第一列的象素才使用该算法。如下是要处理的一些例外: 不引用任何其他象素非常精细地编码一个块的第一个象素, 在第一行中的象素使用象素值 a 作为该预测值, 在第一列中的象素使用象素值 b 作

35

为该预测值。这一算法基本上是试图监测一个边缘。在第一情况下,提示象素 c 和 b 之间以及象素 a 和 x 之间的一个垂直边缘。从而 b 是最好的预测值。在第二情况下,提示象素 a 和 c 之间以及象素 b 和 x 之间的一个水平边缘。从而 a 是最好的预测值。在第三种情况下,没有发现明显的边缘。在这一情况下, a 和 b 都是同样好的预测值,因此使用它们的平均值。

图 20 的压缩网络量化作为 DPCM 处理的结果的差(残留)象素值。图 20 使用具有两个预定的延迟和并行而不是串行处理的一个交错的 DPCM 环。图 20 所示出的电路对应于图 1 的再压缩器 40 所使用的用于处理图 5 所示出的序列中的所交错的象素数据 A 和 C 的电路。压缩器 42 使用一个类似的电路用于压缩所交错的象素数据 B 和 D。由于图 20 的网络压缩一个残留值,在相应的下一块的相应的位置的象素出现之前,该预测器环必须完成一个给定块的一个象素的处理。该交错的象素块数据独立地运动通过该电路,在具有不同速率的输入和输出数据的可变长编码系统中这是重要的。

在图 20 中,单元 230 将来自各个交错的象素块 A, C...(图 5)的交错的象素数据序列 a, c, a, c, ... 延迟一个象素。将一个要压缩的给定的象素值作用到一个减法组合器 210 的非反相(+)输入端。组合器 210 的反相(-)输入端接收来自预测器 215 的预测的象素值。分别由单元 220 和 222 对从组合器 210 输出的残留(差)象素值进行量化和逆量化。在这一例子中,单元 220 提供的量化是固定的,并且保证一个理想的固定数量的数据压缩。单元 230, 232, 234, 236, 238, 240 和 242 是由 54MHz CLK 信号同步定时的寄存器(例如触发器)。单元 230, 232, 240 和 242(Z^{-1})产生一个时钟周期的延时。由于数据交错,要花费两个时钟才前进一个象素。单元 238, 234 和 236 分别产生两个,六个和八个时钟周期延迟,其作为对两个交错的象素块的数据流进行网络处理的结果。逆量化器 222 的输出逼近量化器的输入,但差别一个由量化误差造成的一个小的 DC 偏差。加法器 228 的输出, Input', 与到组合器 220 的输入信号相差这一同样的量。在图 27 中相对于图 20 的所选择的电路单元,示出在几个时钟周期上的交错的输入象素序列 a, c, ... 的时序关系,下面将进行详细的讨论。

图 20 的网络也包括一个加法器 228 和多路转换器 225 和 235, 如图所示。这些多路转换器包括与预测器逻辑 215 和象素延迟单元 234, 236 和 238 相关的象素预测网络。多路转换器 235 的切换状态由所施加的亮度(Y)和色度(C)控制信号确定。Y, C 控制信号是作为如图 3 所示的亮度和色度块交错的函数产生的。控制信号 Y, C 的状态由象素计数/定时过

程得到,使得在宏块亮度象素之后的序列中处理色度象素。对于 8X8 亮度块处理或 4X8 色度块处理,使用 Y 和 C 控制信号来控制预测器电路的适当的延迟量。在将一个“1”逻辑电平控制信号施加到多路转换器 235 时,启动色度象素的处理,使得多路转换器 235 将出现在其“1”输入端上的数据传送。在将一个“0”逻辑电平控制信号施加到多路转换器 235 时,启动亮度象素数据的处理,使得多路转换器 235 将来自延迟单元 236 的输出端施加在其“0”输入端上的数据传送。在一个 8X8 亮度块的情况下,“x”预测器象素是离去的 8 个象素。多路转换器 235 切换输入延迟路径以产生这一更大的延迟。

10 在量化器 220 的输出端产生所压缩(量化)的残留象素输出数据。这一所压缩的数据(图 6)在进行包括 Huffman 编码的进一步的处理之前由单元 242 延迟一个时钟周期。

要特别提到两个触发器延迟单元 232 和 240。使用单元 232 和 240 产生双延迟路径 $\Delta 1$ 和 $\Delta 2$,并且允许相邻象素的预测而不是每隔一个象素。延迟路径 $\Delta 1$ 包括在延迟单元 232 的输出端和延迟单元 240 的输入端之间的电路部件。延迟路径 $\Delta 2$ 包括在延迟单元 240 的输出端和延迟单元 232 的输入端之间的电路部件。每一个延迟单元 232 和 240 代表一个大约 18 纳秒的时钟延迟,或一个 54MHz 的时钟周期。采用这一装置,在就要压缩的象素被时钟同步到该电路的时候,将一个所压缩的输出象素同步出该电路。从而对于要压缩的每个输入象素实时产生一个所压缩的输出象素。

20 在另一个系统中,上述讨论的原则可用于 4 倍的交错,即 4 个数据流水线和四个延迟路径,而不是图 20 的系统中的两个延迟路径。可以将关键的 处理环路分成四个部分以方便同步,这可允许使用一个更快的时钟。也是在这一情况下,使用一个共享的查找表可节约集成电路芯片的面积。在这一例子中,尽管交错该输入象素块,但是在所有的系统中不必交错该输入数据。

使用双延迟路径 $\Delta 1$ 和 $\Delta 2$ 便利了设计所需的整个网络延迟,即在这一例子中的大约 18 纳秒的延迟。在这一方面,要注意到与每个延迟路径有关的广泛的信号处理提供了用于设计延迟的各种手段。由两个延迟路径提供延迟不是关键。最好围绕该时钟优化该电路,使得每个延迟提供大约 54MHz 时钟的一个时钟周期。然而,在另一个系统中,相对于一个给定的电路设计该时钟周期可以是适当的,例如产生不规则的或非周期的时钟周期。这两个延迟路径不须表现出相同的信号处理延迟,但是在所揭示的系统中大约相同的延迟是最好的。

对于信号处理延迟可以优化诸如 $\Delta 1$ 和 $\Delta 2$ 的两个信号处理路径, 其比一个总的路径容易得多, 诸如在没有单元 232 和 240 的情况下。在由单元 232 和 240 所定义的两个路径的情况下, 每个路径可以开始操作而不必等待另一路径的结果。在单个路径的系统的情况下, 每一象素值(例如在单元 234, 215 和 225 输入端的节点的象素值)必须由几个功能处理, 其包括预测器逻辑, 加法器, 量化和逆量化, 并且在下一时钟周期开始之前在该路径的末端出现。另外, 这种象素值必须在这一时间稳定。这是一个严重的限制, 对于所揭示的多个路径系统这种限制是不存在的, 其显示出较多的灵活性。

图 20 的装置在以硬件实现在集成电路中时, 能够以 54MHz 时钟速率实时地对于每一象素输入产生一个压缩的象素输出。图 20 的装置对于设计信号的处理延迟提供了较多的自由, 并且以明显较少的集成电路的表面积产生同样的结果。此外, 所减少的表面积表现出较少的电容, 操作速度更快, 并且消耗较少的功率。使用较快的时钟也是可能的。在这种情况下, 交错仍然带来集成电路面积减少的益处(例如较少的压缩单元和有关的支持单元), 以及使用自动设计工具更好地进行系统优化的优点。

所有逻辑门必须同时以一个时钟合成(synthesized)。在该集成电路设计涉及使用 VHDL 高级语言代码(该语言是公知的), 由该语言代码合成这些逻辑门时, 如所讨论的使用两个延迟路径极大地简化了用于压缩器和解压缩器网络的逻辑门的合成。采用两个延迟路径, 自动逻辑设计很快收敛, 使得可以更快, 更精确和更容易复制地合成这些逻辑门。

除了方便了更容易复制的设计, 图 20 的所描述的双处理路径促进了使用交错技术以产生更好的带宽和使用共享的逻辑单元(例如查找表)。这种双处理路径也便利了将该设计划分成特定系统所要求的功能单元或模块, 诸如在这一实施例中的预测和压缩计算功能。这种模块可以按需要进行裁剪以适合特定系统设计的需要。

要注意的是, 考虑到所交错的压缩器操作, 已发现最好采用一个快速时钟的两个周期将一个压缩器用于所交错的数据, 而不是使用两个压缩器电路, 每个电路以该给定时钟的一半进行时钟同步。通过交错使用一个时钟的两个周期便利了时序优化, 如所讨论的, 并且交错允许处理两倍的数据。在所揭示的系统中, 在一个时钟周期进行一个给定象素值的预测, 同时在下一个时钟周期进行对该象素的计算(诸如量化和逆量化)。比如, 对于所交错的象素块 A 和 C, 在一个 54MHz 的时钟周期期间预测来自块 A 的象素数据, 同时对来自块 C 的象素数据进行量化计算。在下一个时钟周期, 对块 A 的象素数据进行量化计算, 同时对块 C 的象素数据进

行预测。因此该系统交替地对不同的交错的块进行预测和计算。使用 54MHz 时钟的两个时钟周期提供了采用硬件电路制造中可用的适当工具以优化电路的延迟的机会。图 27 示出了交替地预测象素值和计算压缩值的过程。

5 利用图 27 示出的该过程,以该 54MHz 的压缩时钟的几个周期处理相关的交错的象素块 A 和 C 的交错的象素“a”和“c”。假设以第一个象素块 A1 的第一个象素 a1 开始该处理。考虑图 27 和图 20,第一个时钟周期使得来自寄存器(触发器)230 的输入端的象素 a1 被同步到其输出端,从而由单元 220 量化象素 a1,并且在其出现在寄存器 232 的输入端之前由
10 单元 222 进行逆量化,所有这些都是第一个时钟周期进行的。在该点上,将象素 a1 表示为象素 a1',因为在该点的象素 a1 逼近输入象素 a1,但有一个小的 DC 偏差,这是由于与单元 220 和 222 的处理相关的量化误差造成的。

第二个时钟周期使得下一个出现的象素,即所交错的象素块 C1 的
15 第一个交错的象素 c1,被以类似于对象素 a1 处理的方式进行处理。另外,第二个时钟周期使得象素 a1'被同步到寄存器 232 的输出端,并且从而同步到包括单元 215 和 225 的预测网络。这使得象素 a1'的预测值出现在寄存器 240 的输入端。从而在第二个时钟周期期间压缩(量化)象素 c1,同时对先前压缩的象素 a1'进行预测处理。

20 在第三个时钟周期,将预测地处理的象素 a1'传送到寄存器 240 的输出端,在单元 210 中进行相减组合,由单元 220 进行压缩,并且作为所压缩的输出值 a1"出现在输出寄存器 242 的输入端。在下一个时钟周期将象素 a1"从这一寄存器同步到随后的霍夫曼编码电路。也是在该第三个时钟周期,在产生压缩的象素值 a1"的同时,由单元 215 对交错的象素 c1'
25 进行预测处理。对交错的块 A 和 C 的其余象素 a_n, c_n 该过程继续进行,从而在每一个时钟周期,分别对交错的象素进行预测和压缩处理。

如果没有数据交错,其处理将不得不在一个时钟周期例如从象素值 a1 前进到输出压缩的值 a1"。这一要求对于速度和时序是一个严重的约束,而通过所描述的交错处理,则可避免这种约束。交错的处理也允许共用
30 量化和逆量化逻辑,使得集成电路的面积和功耗减少。

可以独立于 MPEG 压缩使用所描述的交错的压缩器操作,并且这也代表在消费者视频应用(例如诸如 VCR 和摄像机(camcorder)的家庭视频系统)中使用的一种经济的处理系统,以提供 25%-50%的预测压缩,从而不需要更复杂和昂贵的 MPEG 压缩。

35 可以以一个更快的时钟使用四个而不是两个交错的象素块。在这种

情况下,可以一次处理四个 8X8 亮度象素块的一个整块。

在所揭示的系统中,如图 21 所示安排每个解压缩器网络。该解压缩器电路类似于图 20 的压缩器电路,除了单元 210 是一个 9 位加法器以及已经去掉单元 220, 222, 228 和 242 之外。路径 $\Delta 2$ 涉及的处理少于路径 $\Delta 1$, 然而即使在路径 $\Delta 2$ 中包括一个 9 位加法器 231 增加了大约 9 毫秒的时间约束, 这使解压缩器的设计复杂化了。考虑与此, 要注意加法器 231 在多路转换器 225 接收到有效数据之前不能开始计算。因此减少环路的时序约束是有利的。使用双延迟路径可实现这些, 并且极大地简化了整个设计。

10 对于解压缩, 已经发现预测路径 $\Delta 1$ 是更重要的路径。已经发现在压缩的情况下, 预测路径 $\Delta 2$ 是更重要的路径, 其中密集的数据处理决定使用更慢的 54MHz 时钟。

如前所说明的, 网络 80-84 中的每个解压缩器独立地操作, 使得快速地分发象素数据。通常通过时钟同步的设备, 诸如触发器或寄存器, 完成数据的传输。在从两个或多个数据源导出要进行流水的数据时, 例如在交错的数据的情况下, 在一些寄存器中的数据来自一个数据源的同时, 其他寄存器的数据来自另一个数据源。响应于一个共同的数据时钟, 数据流到一起, 但在连续的寄存器中的数据是相互独立的。在两个数据源同步开始和停止的时候, 只要该数据流(流水线)在同一时间开始和停止, 可以避免系统操作的问题。

20 在一个数据源停止发送数据, 而同时另一个数据源继续发送数据的时候, 会出现一个问题。在诸如 HDTV 信号处理器的数据密集的流水线中, 对于产生一个精确的高质量显示图象, 每秒大量的计算是关键。只要有一个或两个(或多个)数据源停止发送数据, 这种系统不能中断该数据流。在这种情况下, 重要的是控制该流水线的数据流, 以在中断另一数据源时维持从一个未中断的数据源提供的输出数据的正确相位。

30 重要的是, 在诸如 HDTV 系统的数据密集的图象处理系统中该数据时钟不被停止。在这种系统中, 诸如压缩和解压缩子系统的某些部件具有不同的输入和输出数据处理要求和不同的数据速率。例如, 用于显示的解压缩的输出象素必须连续地输出, 因此要求一个连续的时钟, 但是在一个时钟不存在的时候, 要解压缩的压缩的输入数据可以以空间间隔(null interval)零散地到达。如果在输入数据不存在的时候该解压缩的时钟被停止, 则解压缩的象素数据时钟同步输出也停止。在数据密集的高清晰度的图象处理和显示系统中这是破坏性的。因此如所要讨论的, 在某些情况下重复数据是有利的, 尤其是在解压缩过程中使用霍夫曼译码

时。

在所揭示的系统中,对压缩网络(图 20)的输出进行霍夫曼编码。霍夫曼译码与在一个译码器的解压缩有关。由于不同的编码的字长度,霍夫曼编码/译码是一个具有不同的输入和输出数据速率的统计过程,因此使用缓冲器适应可变的数据内容。

如参照图 23 和 24 所看到的,在由一个寄存器的流水线序列处理来自不同的数据源的数据时,使用来自每隔一个寄存器的反馈保持一个(来自第一个数据源的)数据分量流经该流水线,同时(来自第二个数据源的)其他数据分量对其自身进行重复。利用这一技术,采用来自数据源的交错的数据,在一个数据源已经停止提供数据的时候,可以以一个需要的预计的速率通过该流水线处理数据。

重复数据等价于停止该数据时钟,但没有启动-停止的同步问题。使用重复的数据最好是不使用数据(即空数据),因为在恢复延时中常常丢失数据周期。重复数据对于保持数据流的完整性是重要的,并且不象不发送数据一样是破坏性的。

该数据重复过程可以在一个 8X8 象素块的期间(64 个时钟周期)重复数据,而不引入系统的复杂性。根据系统的性质以及有关的处理,较长的重复延迟也是可能的。比如,在象素预测处理期间,多达 6 个块将存储在存储器中。在这种情况下,在一个数据源被破坏的情况下,一个块可被有效地保持(重复),而同时从存储器中获取其他的块。预计可容许在 1 个或 2 个宏块的间隔上重复延迟。

在出现一个数据源被破坏的情况下,重复数据最好是加入空数据,因为处理空数据是低效率的。象其他数据一样,空数据存储在存储器中,从一个空数据情况下恢复是浪费时钟周期的,即除去空数据之后重新加载有效数据。在诸如高清晰度电视系统的数据密集系统中这是重要的考虑,其中存储器带宽是非常重要的,应当尽可能地减少数据处理所要求的时钟周期数目。

在图 23 中,输入流水线分别从各个数据源,例如从诸如图 1 的帧存储器 60 的存储器中的各个单元,传送数据 X 和 Y。数据 X 和 Y 是相互独立的,并且没有特定的顺序,即它们可以是或不是交错的,并且表示从存储器请求的任一象素数据。在这一例子中,X 和 Y 数据分别表示由运动补偿网络(图 1)使用的前进和后退运动信息。即使 Y 数据不存在,也必须处理 X 数据,反之亦然。

图 23 的电路从存储器 60 传送数据到图 1 的解压缩器网络 80-84,并且很好地适应 MPEG 处理。可产生一个 MPEG 编码的 P 或 B 图形,但可能

发生一个给定的宏块可能不具有一个或其他用于处理的前进或后退数据。图 23 的装置识别这一可能性。

5 对于一个数据块,输入 FIFO 缓冲器 332 和 334 与每一个输入有关。在这一例子中,缓冲器 332 和 334 表示图 1 的每一解压缩器 80,82 和 84 的输入缓冲器。每一缓冲器在适当时候(例如响应涉及单向对双向预测的处理)发送一个信号 Req,通过存储器控制器 134 从存储器请求数据,并且接收表示一个数据可用于发送的一个返回确认信号 Ackn。如所公知的,数据读取/写入控制信号控制存储器源和缓冲器之间的数据流动。

10 响应 CLK/2 数据时钟,通过多路转换器 336 在一个共用的数据线上对输入 X,Y 数据进行多路转换,在多路转换器 336 的输出端产生一个交替的 X,Y 数据的流水线。通过一串行的反馈寄存器部分 360 和 364 处理来自多路转换器 336 的数据。所使用的寄存器部分的数量是交错的数据单元的数量的函数,在这一情况中,是两个寄存器部分。部分 360 包括一个输入多路转换器 338 和串联的寄存器(触发器)340 和 342,如图所示
15 的。每一个寄存器单元由 CLK 信号以 81MHz 进行同步。寄存器部分 364 也类似地安排。最后一个寄存器单元 350 的输出施加到图 1 的数据解压缩网络,后者解压缩包括前进和后退的运动预测信息的数据。一旦该解压缩器接收到数据,将对该数据解压缩。解压缩器一直等待,直到填充了 X,Y 缓冲器。在每一个寄存器部分,从该寄存器部分的最后一个寄存器的
20 输出端反馈到该相关的多路转换器的切换控制输入端,即从寄存器 342 的输出端到多路转换器 338 的“1”输入端。由反馈寄存器部分 360 和 364 构成的网络作为一个选择性的具有两种工作模式的数字取样和保持网络进行工作。在一种模式中,对数据进行取样和保持以产生数据重复操作。在另一个模式,正常地没有重复地传输数据。

25 单元 356,例如数字比较器,读取 Req 和 Ackn 信号线的状态。如果一个 FIFO 缓冲器产生一个 Req 信号,并且没有从该存储器源接收到一个返回 Ackn,则单元 356 产生一个为“1”电平或“1”状态的数据 Halt(停止)信号。在该 Halt 信号为“0”状态时,数据正常流经该流水线,但是在 Halt 信号是“1”状态时,将如下面所解释的,对数据进行重复。在没有从一个
30 特定的输入端接收到一个 Ackn 信号时,该 Halt 信号使得在每一个寄存器部分重复或循环该最后一个有效数据分量。这通过图 24 的波形而示出,如下面将要讨论的。如果没有从 X 和 Y 输入数据源接收到一个 Ackn 信号,停止该时钟,并且没有数据循环。

35 从而在该 Halt 信号为 0 电平时,数据正常流经该流水线,以正确的交错的(时钟)相位关系保持输入数据 X 和 Y,使得该同步时钟输出的数据在

数据源 X 数据和数据源 Y 数据之间交替。这一相位关系对于防止混合数据是重要的。在这种情况下,每一寄存器部分的输出数据(例如在寄存器 342 和 350 的输出端)对应于两个时钟前的输入数据(即输出=输入(Z^{-2}))--Output=Input(Z^{-2})。该 Halt 信号为 1 电平时,该相关的多路转换器 (338 或 344)将该输入信号从该输出端解耦合,使得每一个寄存器部分简单地循环数据。这些操作状态由图 25 和 26 分别示出。

图 24 示出了这样一种状态,例如只有在来自 Y 数据的数据源的某些数据已停止时,该 Halt 信号才为 1 电平。在该 Halt 信号为活跃的同时,循环(重复)使用该 Y 数据,直到该 Halt 信号返回到通常的 0 电平,并且 Y 数据再次流动。在这一期间,来自数据源 X 的数据无中断地流动。在图 24 中,该输入波形包含 X 和 Y 数据分量的一个交错的序列。在这一例子中,对于接着分量 X2 的 Y2 分量还没有接收到一个 Ackn 信号。因此,单元 356 的 Halt 信号的通常的“0”状态转换为“1”状态,使得每一个寄存器部分 360 和 364 重复该最后一个有效的 Y 分量,在这种情况下,是 Y1 分量,只要该 Halt 信号为“1”状态。该 Halt 信号连接到多路转换器 238 和 244 的控制输入端,从而该 Halt 信号的“1”状态使得每一个多路转换器传送连接到其“1”切换输入端的信号,在这种情况下,是 Y 数据分量。

输入信号波形的阴影部分表示该丢失的 Y2 分量,即在分量 Y1 之后,该第二数据源不发布 Y 分量。对于三个 Req/Ackn 周期重复该 Y1 分量,据此产生三个 Halt 信号,并且重复分量 Y1 三次,如在图 24 的输出波形中所示出的。以后,对于分量 Y2,第二数据源产生一个 Ackn 信号,其出现在跟随数据分量 X5 的输出波形序列中。

该 Halt 信号也被提供给所相关的解压缩器网络的控制输入端,以指示该解压缩器忽略该数据流中所重复的数据。如前面所描述的,交错的 X, Y 数据分量是独立的,不需要在任一特定(数字)的序列中彼此相随。需要的只是与一个给定的输入相关的数据沿着一个规定的序列,例如 X5 跟着 X4, X4 跟着 X3, X3 跟着 X2, 等等。而例如 Y2 跟着 X5 是没有什么后果的。

图 28 描述了图 23 的安排为用于并行操作的网络。将来自多路转换器 336(图 23)的交错的输入数据通过多路转换器 285 和 286 提供给并行的寄存器 280 和 282,寄存器 280 和 282 的输出通过多路转换器 284 被多路转换到一个数据输出路径。该多路转换器 284, 285 和 286 的操作由 Halt 1 和 Halt 2 控制信号进行控制,该控制信号与各个数据源有关,并且产生如结合图 23 所描述的效果。

35

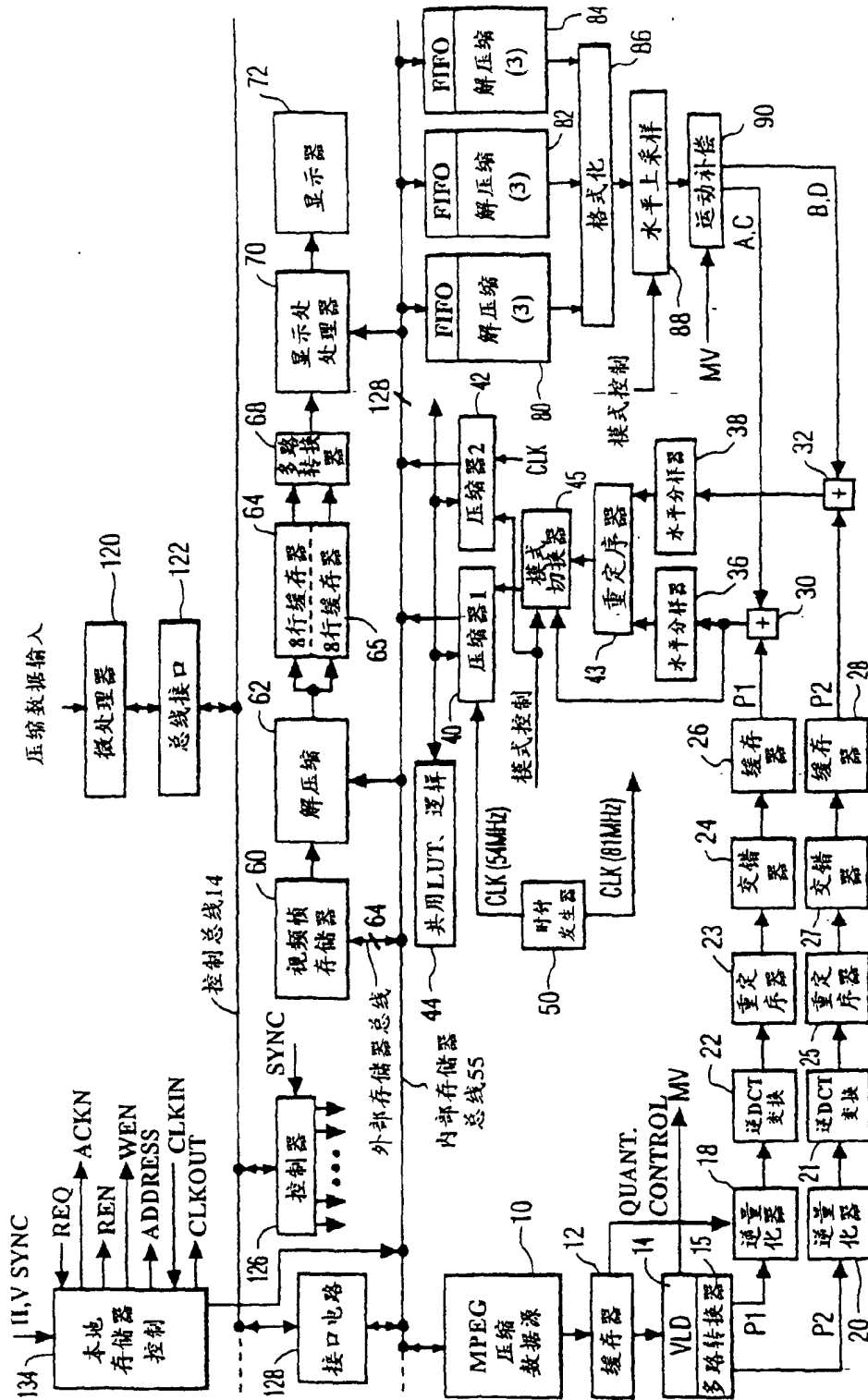


图 1

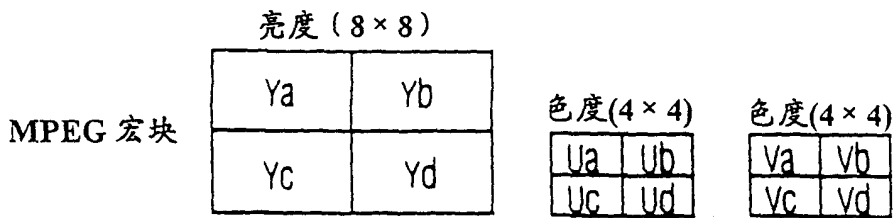


图 2

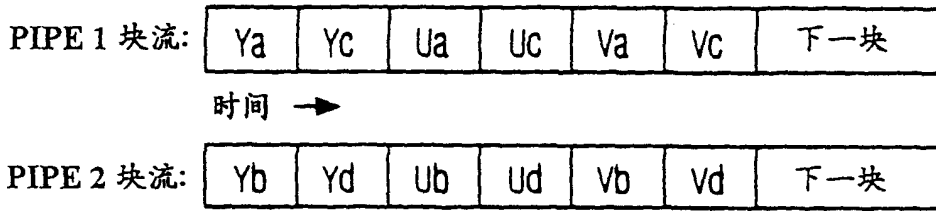


图 3

流水线中的U色度块数据

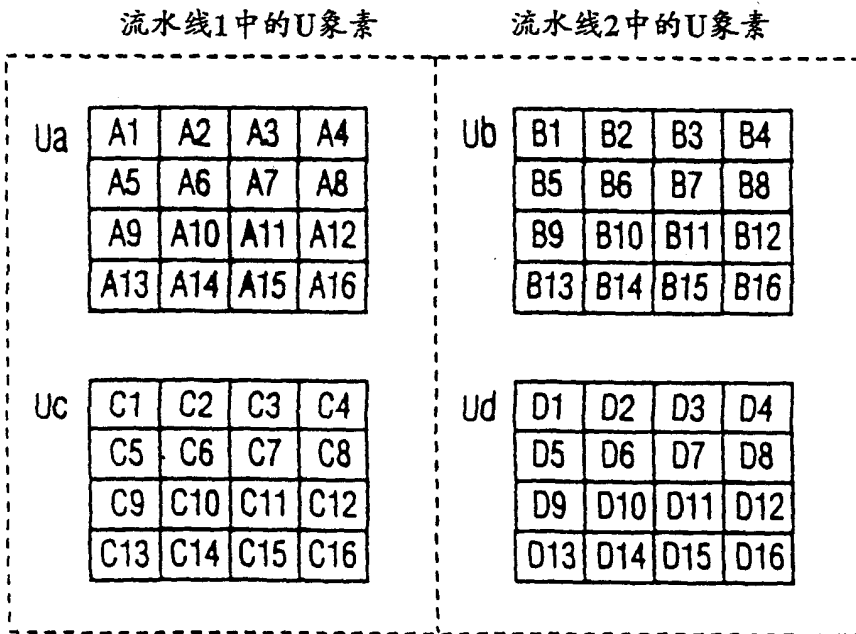


图 4

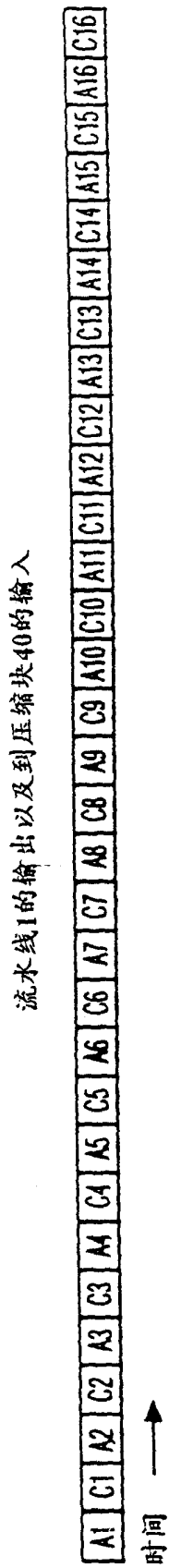


图 5

与流水线1有关的从压缩块40的输出



图 6

在去除交错后发送到外部存储器的数据块

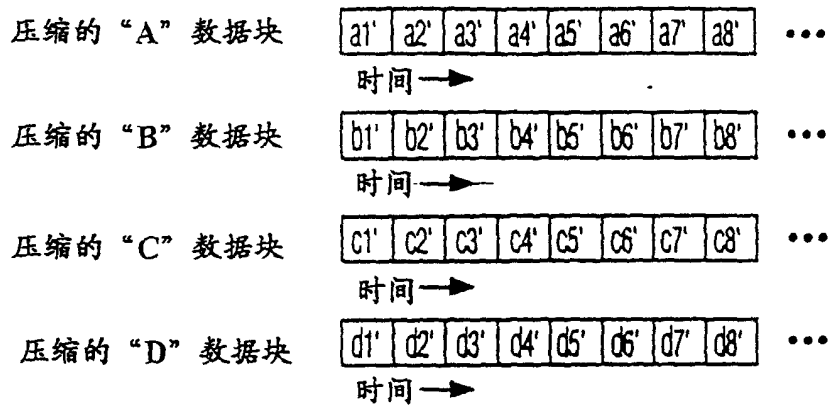


图 7

准备用于解压缩的两个交错的压缩的数据块

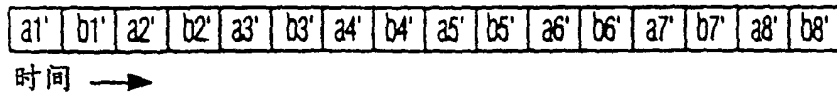


图 8

具有两个恢复的仍然是交错的数据块的解压缩输出

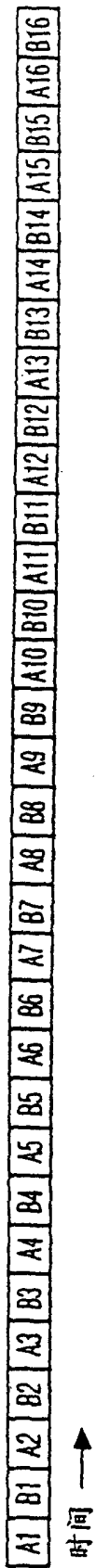


图 9

在显示块中经过未来处理之后的象素的显示顺序

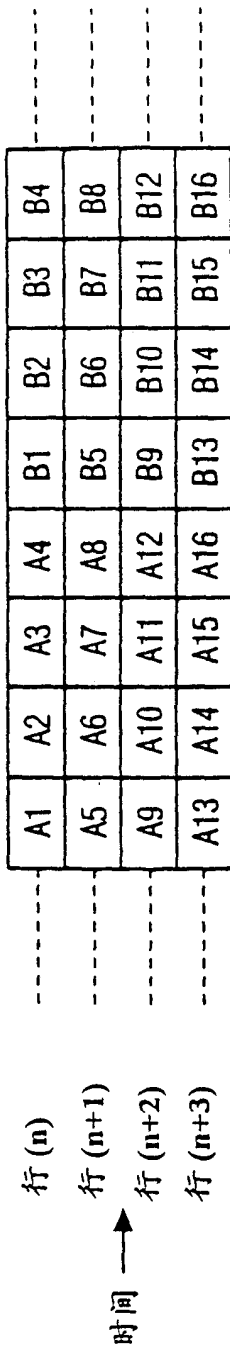


图 10

流水线1的输出和到过滤波器36的输入

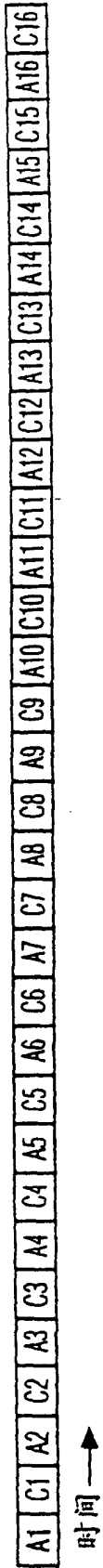


图 11

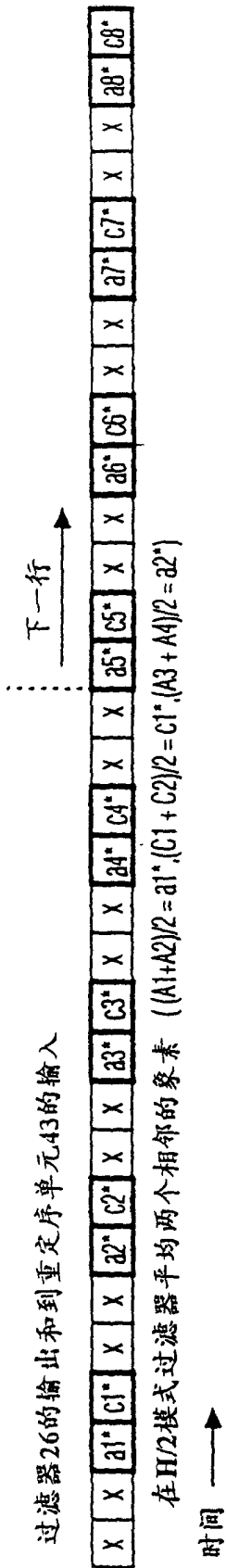


图 12

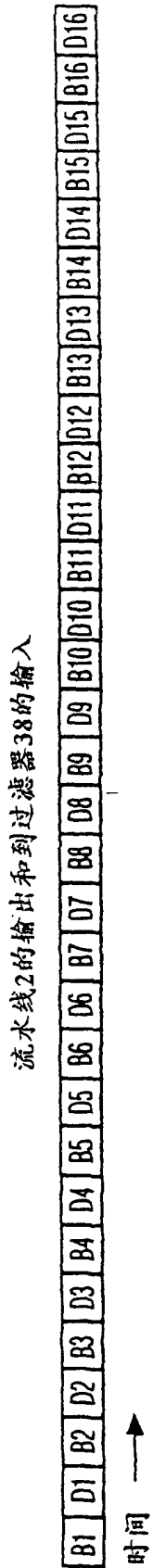


图 13

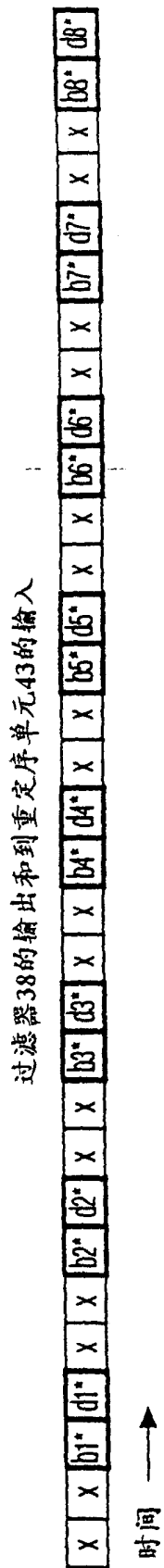


图 14

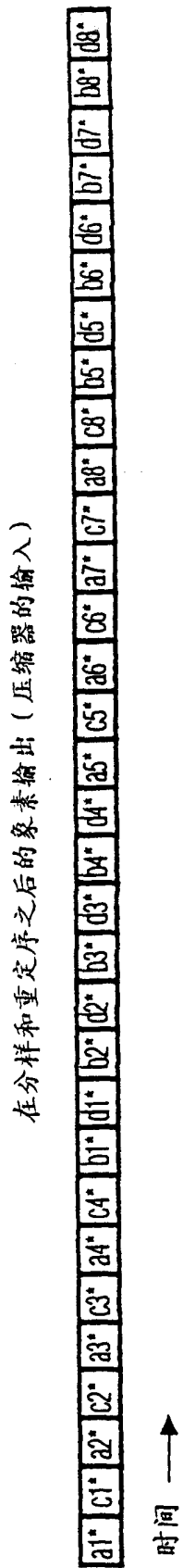


图 15

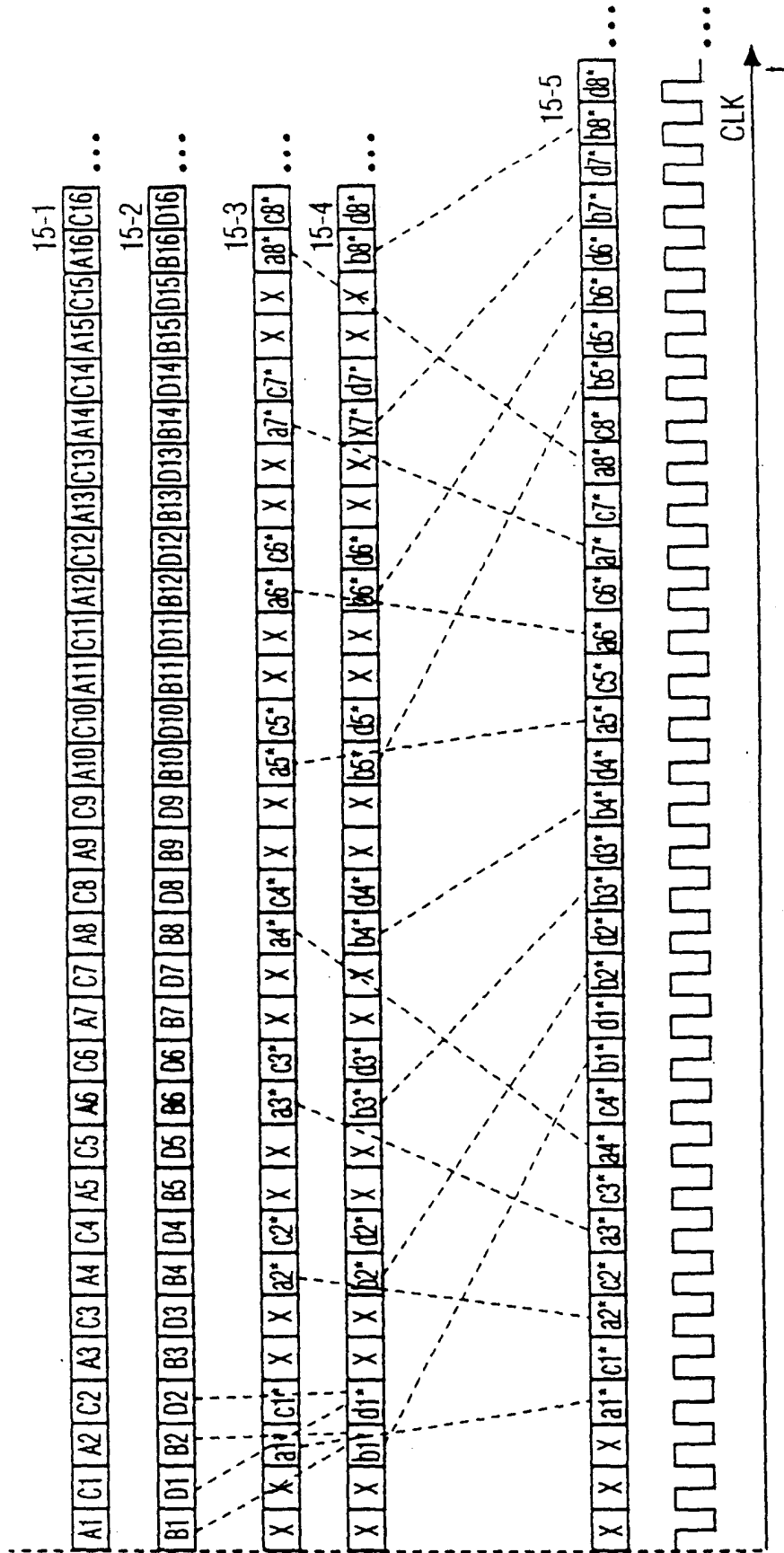


图 15A

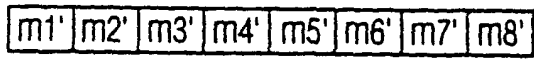
从与流水线1有关的压缩共40的输出



图 16

发送到外部存储器的压缩的数据

来自A和B的块



来自C和D的块

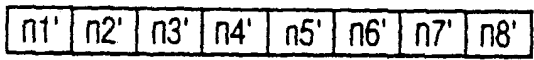


图 17

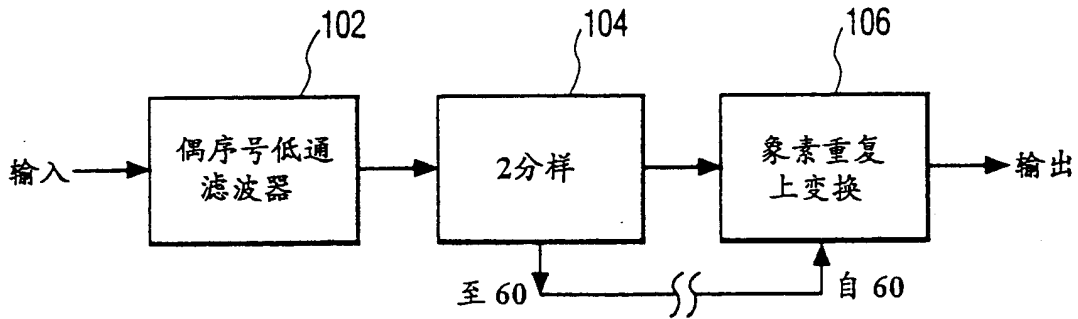
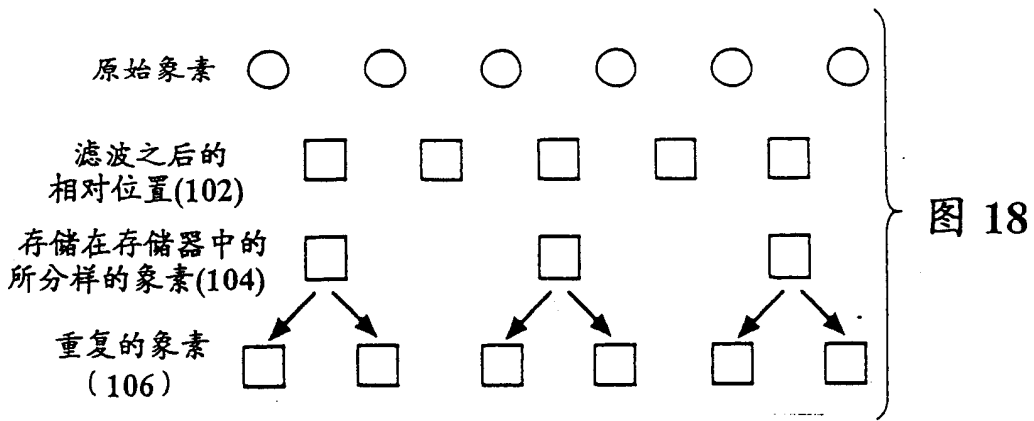


图 19

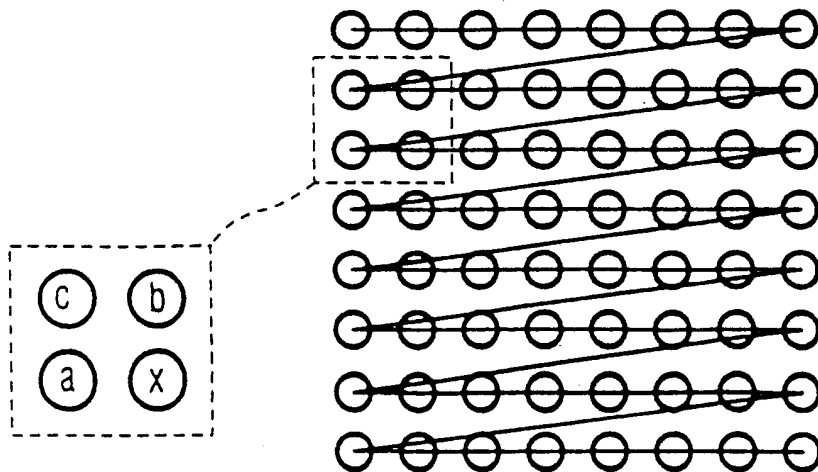
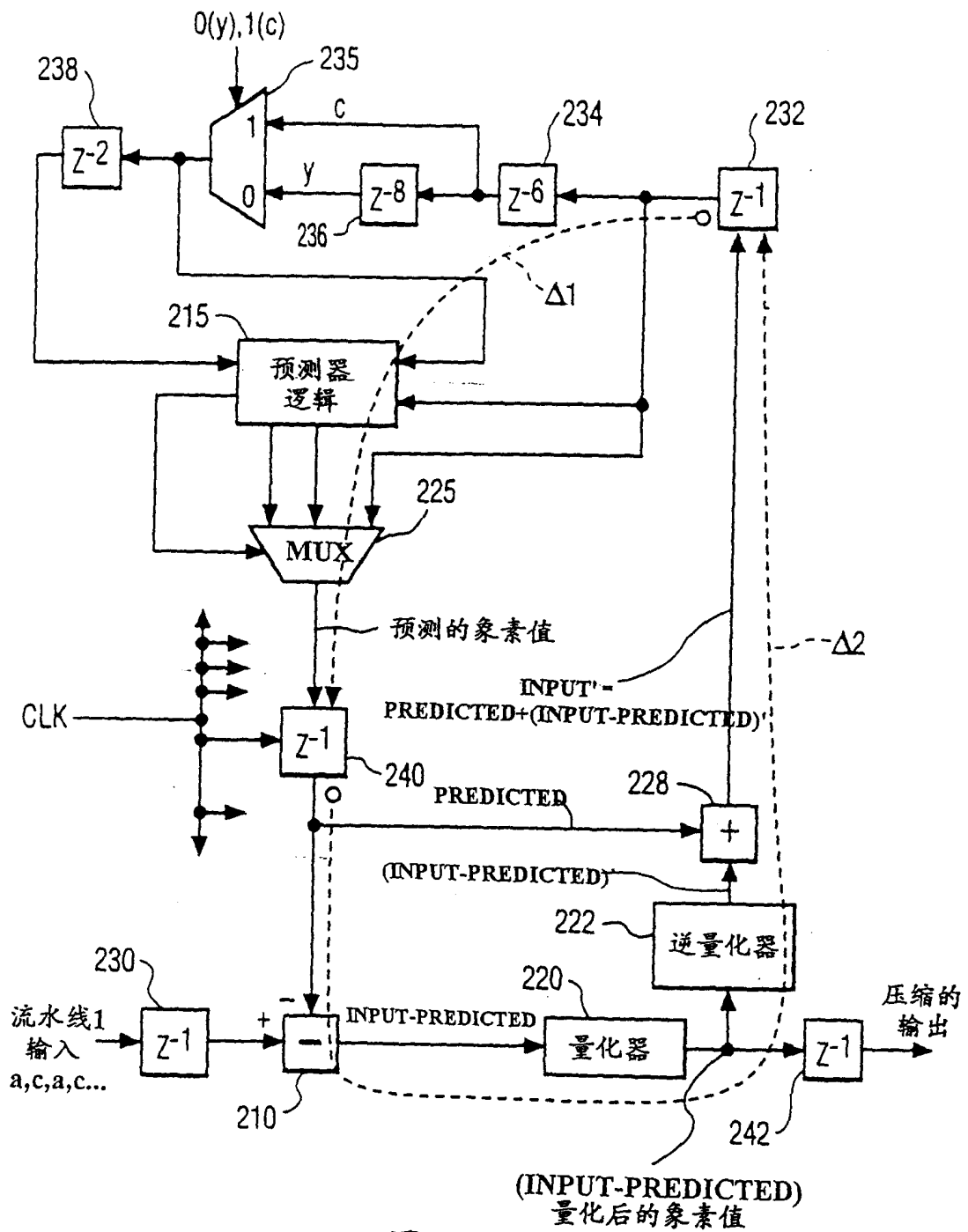


图 22



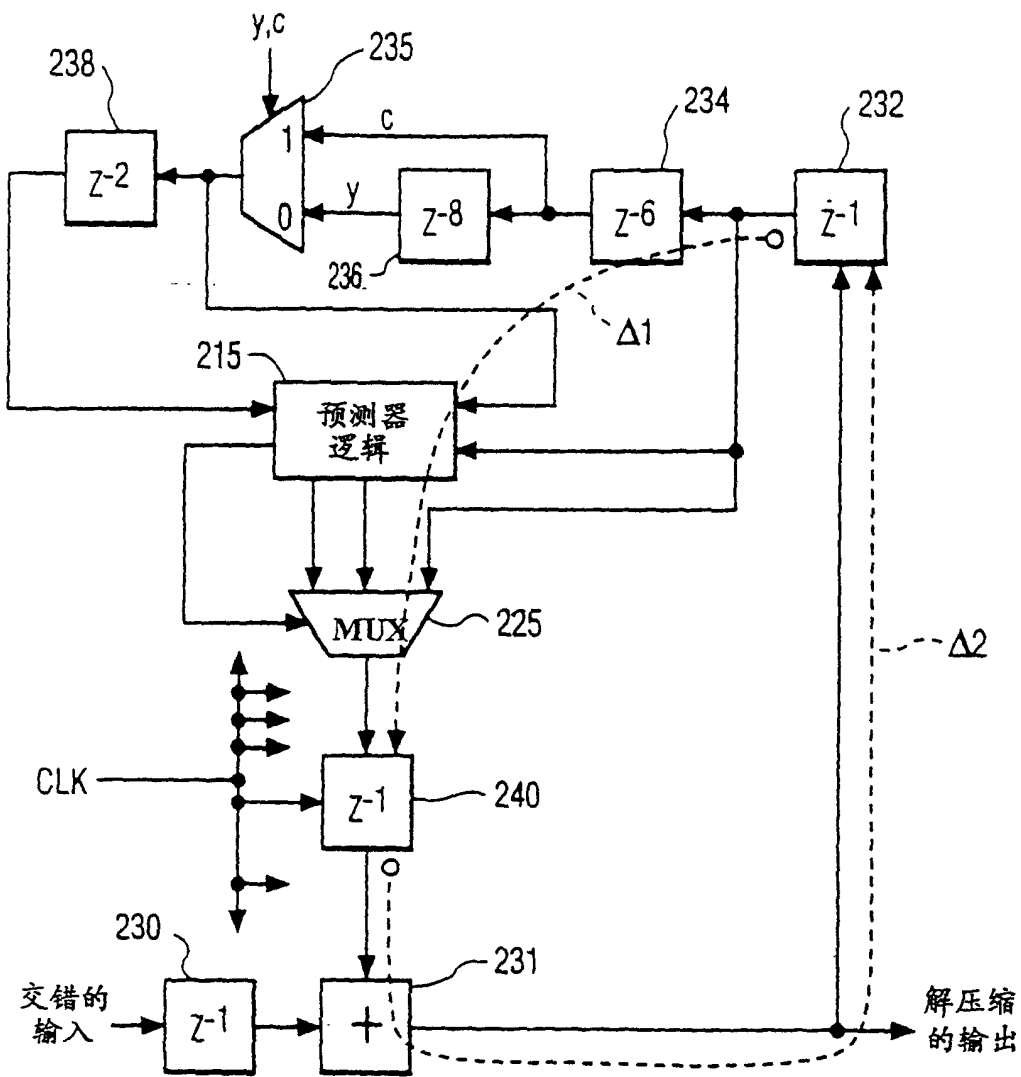
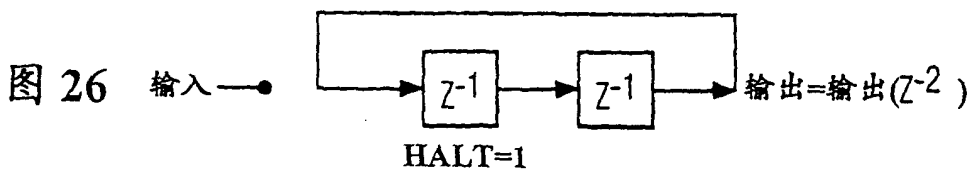
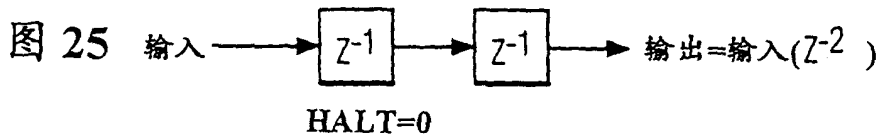
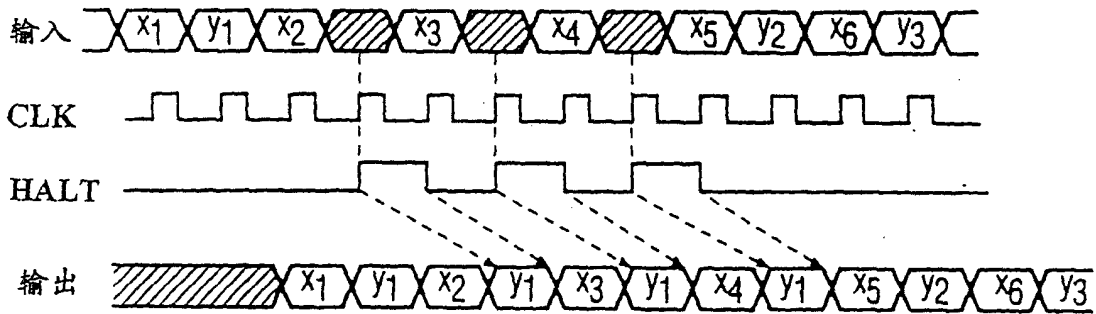
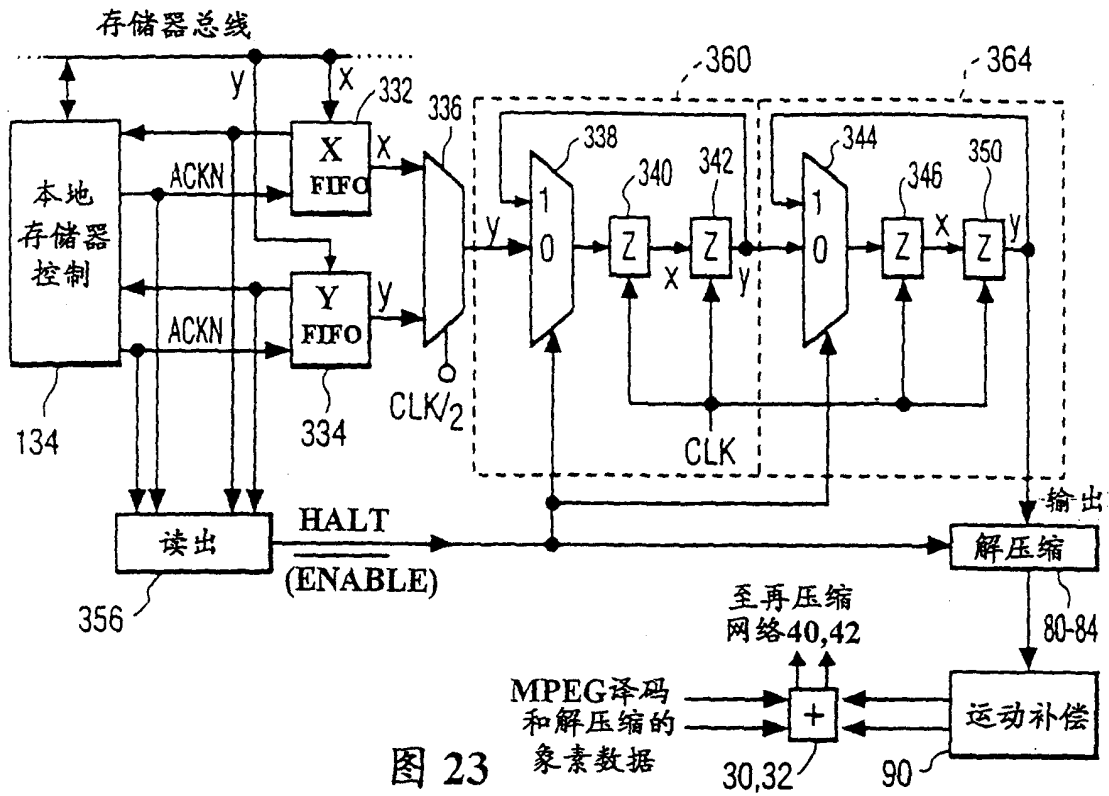


图 21



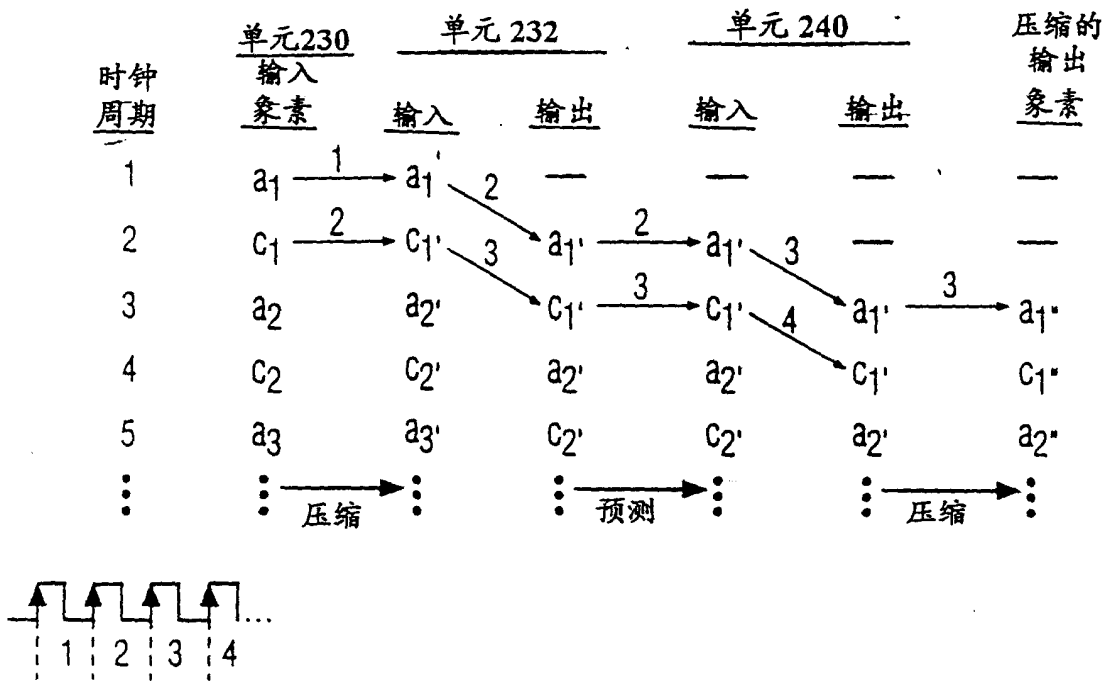


图 27

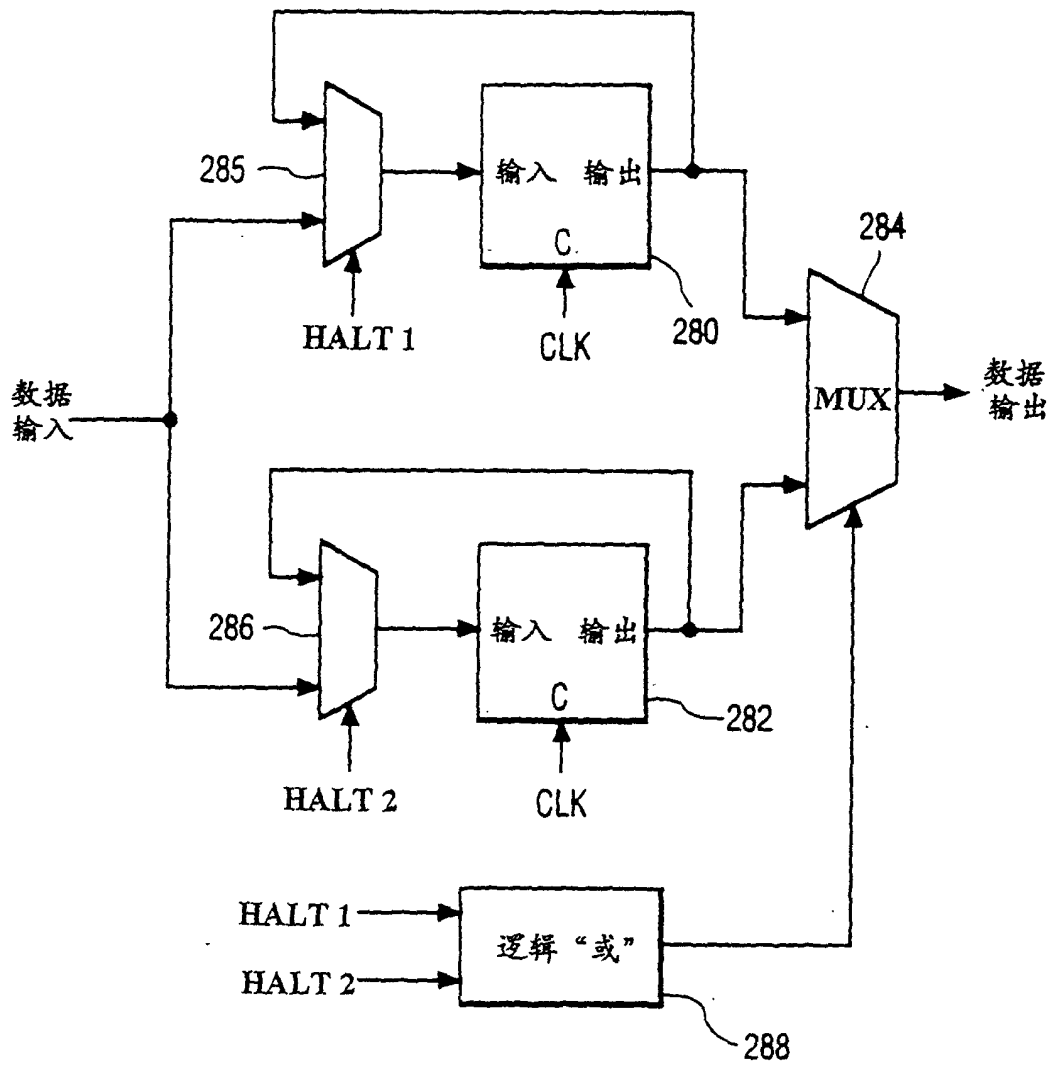


图 28