[19] 中华人民共和国国家知识产权局

[51] Int. Cl[7]
G06F 3/12

[12] 发明专利说明书

[21] ZL 专利号 94105565.5

[43] 授权公告日 2003 年 2 月 26 日

[11] 授权公告号 CN 1102257C

[22] 申请日 1994.4.15 [21] 申请号 94105565.5

[30] 优先权

[32] 1993. 4.15 [33] JP [31] 88369/1993

[32] 1993.11.16 [33] JP [31] 287079/1993

[71] 专利权人 佳能株式会社

地址 日本东京

[72] 发明人 鬼冢康如 镰田雅史

审查员 张静海

[74] 专利代理机构 中国国际贸易促进委员会专利商标事务所

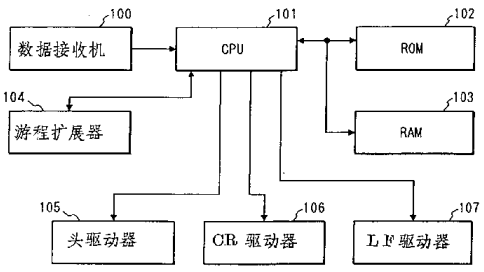
代理人 王以平

权利要求书 2 页 说明书 21 页 附图 19 页

[54] 发明名称 输出装置和方法

[57] 摘要

通过预测函数将数据转换。转换的数据被游程压缩，并且产生的压缩字形数据存储在 ROM 中。当输出时，压缩数据传送到游程扩展器。该游程扩展器产生字形的位，并且顺序存储在 RAM 中。该存储的数据传送到反向预测函数处理器，并且形成字形图形。



ISSN1008-4274

1．一种输出装置，包括：

用于存储具有白点的字形数据的存储装置（102），该字形数据已经通过预测函数过程被增大了，其中，将具有一个操作数部分和一个数据部分的第一代码赋予所述字形数据中所包含的白数据，将指示黑色位的数量的第二代码赋予该字形数据中包含的黑数据；

字形数据产生装置（101），用于使扩展装置（104）扩展存储在所述存储装置中的字形数据，使RAM装置(103)存储扩展了的字形数据，使反向预测处理装置(108)在存储于所述RAM装置中的字形数据上进行反向预测处理；以及

记录装置（105、106、107），用于根据由所述字形数据产生装置产生的字形数据记录一个字符，

其中，所述扩展装置包括：

加法器装置（207），用于根据存储在所述存储装置中的字形数据的第一代码把所述数据部分的值加到与所述操作数部分对应的起始值上，从而获得连续的白点的位的数量；

白数据位图形转换器（212），用于在该白色位的数量小于一个字中剩余位的数量时，按由所述加法器装置获得的白色位的数量转换所述字形数据；

黑数据位图形转换器（213），用于按根据存储在所述存储装置中的字形数据的第二代码获得的连续黑点的位的数量转换所述字形数据；

锁存装置（216），用于锁存由所述白数据位图形转换装置转换过的字形数据或由所述黑数据位图形转换装置转换的字形数据，以及

白字计数装置（230），用于在由所述加法器装置获得的白色位的数量等于或大于一个字中剩余位的数量时，输出通过从白位的数中减去剩余位的数量、并把该减法的结果除以形成一个字的位的数量而得

到的商数，将其作为将由所述字形数据产生装置读出的白字的数量，并由所述锁存装置锁存余数，以及

其中，所述字形数据产生装置读由所述锁存装置锁存的所述字形数据的一个字，并将其传送到 RAM 装置中，并根据由所述白字计数装置输出的白字的数量把该数量的白数据传送到 RAM 装置。

2. 如权利要求 1 的输出装置，其中，所述形成一个字的位的数量是 8 位、16 位或 32 位。

3. 如权利要求 1 的输出装置，其中，所述字包括字节或长的字。

4. 如权利要求 1 的输出装置，其中，所述记录装置用记录头在记录介质上记录一个字符。

输出装置和方法

本发明涉及使用游程压缩的字形输出字符的输出装置和方法。

迄今，在记录装置中，象字形数据或类似数据的图象数据存储于记录装置中的记录区域中 (ROM)，而且对应输入数据的字符位图形由记录区域存取，并且被转换成打印数据和打印出来。在近几年来，关于象字形数据或类似数据的图像数据、较高清晰度的打印数据的需求是为了提高字符的质量。这就是说构成字形数据一个字符的点数的增加正比例于清晰度，由 (48点 ×48点) 形成的字形需要每字形288个字节 (2306比特)。在是日本字形的情况下，现在假设在一种字形中有大约7000个字符，存储区域ROM的容量大约需要2M字节。迄今，一种压缩图象数据和将其存储于记录装置中的方法被用来减少ROM的容量。例如有一种游程压缩的方法。

迄今，在记录装置中，假如打印的宽度大于使用具有预定宽度的打印纸的宽度，或者假如根据切页纸的规格打印连续的纸页，这种打印使用了压缩的形式。具体地说，在垂直方向相对较长的A4规格的纸上打印相当于两页其规格为15英寸 ×11英寸的连续纸页的图象数据的情况下，其形状缩小为1/2。在横向较长的A4规格的纸上打印相当于一页 (15英寸 ×11英寸) 规格的连续纸页的图象数据的情况下，该形状缩小为2/3。如果在横向较长的A4规格的纸上打印与一页横向较长的B4规格的纸对应的图象数据，该图形缩小为

4/5。倘若在垂直方向上较长的A4规格的一张纸上打印与一页垂直方向较长的B4规格纸所对应的图象数据，该图形缩小为7/8。照这样，即使是所设计的记录装置为的是在A4规格的切页纸上打印，记录器也能处理比A4规格纸要宽的纸。

上述通常的装置 (1) 具有如下的缺点。

1. 当字体没有压缩，但是已存储在记录装置中时，记录区域的记录容量是非常大的，导致了成本的显著增加。

2. 如果使用游程压缩方法数据压缩一种字体，压缩效率没能完全地提高。

3. 倘若压缩字体并且将其存储于记录装置的存储区域中，由于扩展压缩数据的逻辑是复杂的，而且在位单元基础上执行处理，扩展处理的软件负荷是相当大的，作为整个记录装置对于输入输出信息的通过是有不利的影响，将导致打印速度的减小。

前面通常装置 (2) 具有下列的缺点

1. 假如在一行图象数据显现以后显现一种字体和缩小图形，垂直方向的缩小必然是在位单元基础上执行，缩小处理的软件的装入工作量是很大的，并且作为整个记录装置而言将对信息的通过量有不好的影响，导致了打印速度的下降。

2. 必须依赖于不同的缩小比率分别地设计软件，因此所需的设计时间是明显地增加，

因此，本发明的一个目的在于提供一种记录装置，在该装置中，当图象数据被压缩和被存储在该记录装置中时，在使用一种压缩方法的游程压缩对原来的字形数据进行预测函数处理的情况下，该数据被转换以致使连续的白点尽可能的长，通过游程压缩的压缩率增

加了，记录区域的ROM容量显著减小，由硬件实现图象数据的进一步的解码处理，因此字形数据的造形处理速度提高了，并且没有减少打印的信息通过量。

本发明的另一个目的是提供一种记录装置，在该记录装置中压缩图象数据并且将其存入该记录装置中，使用游程压缩作为压缩方法，另外由硬件实现数据扩展，同时还能形成和记录图象数据。

本发明还有一个目的是提供一种记录装置，其中用一个简单的硬件实现垂直方向的缩小并且形成和记录图形的缩小数据。

本发明另一个目的在于提供一种记录装置，其中用一种改进的硬件作行程扩展处理，以高速形成打印字形，从而获得较高的打印速度。

为了实现上述目的，根据本发明提供了一种记录装置以便将来自外部的字符数据转换为内部的字形，并且作为图象数据记录下来。本发明的一种输出装置，包括：用于存储具有白点的字形数据的存储装置，该字形数据已经通过预测函数过程被增大了，其中，将具有一个操作数部分和一个数据部分的第一代码赋予所述字形数据中所包含的白数据，将指示黑色位的数量的第二代码赋予该字形数据中包含的黑数据；字形数据产生装置，用于使扩展装置扩展存储在所述存储装置中的字形数据，使RAM装置存储扩展了的字形数据，使反向预测处理装置在存储于所述RAM装置中的字形数据上进行反向预测处理；以及记录装置，用于根据由所述字形数据产生装置产生的字形数据记录一个字符，其中，所述扩展装置包括：加法器装置，用于根据存储在所述存储装置中的字形数据的第一代码把所述数据部分的值加到与所述操作数部分对应的起始值上，从而获得连续的白点的位的数量；白数据位图形转换器，用于在该白色位的数量小于一个字中剩余位的数量时，按由所述加法器装置获得的白色位的数量转换所述字形数据；黑数据位图形转换器，用于按根据存储在

所述存储装置中的字形数据的第二代码获得的连续黑点的位的数量转换所述字形数据；锁存装置，用于锁存由所述白数据位图形转换装置转换过的字形数据或由所述黑数据位图形转换装置转换的字形数据，以及白字计数装置，用于在由所述加法器装置获得的白色位的数量等于或大于一个字中剩余位的数量时，输出通过从白位的数中减去剩余位的数量把该减法的结果除以形成一个字的位的数量而得到的商数，将其作为将由所述字形数据产生装置读出的白字的数量，并由所述锁存装置锁存余数，以及其中，所述字形数据产生装置读由所述锁存装置锁存的所述字形数据的一个字，并将其传送到 RAM 装置中，并根据由所述白字计数装置输出的白字的数量把该数量的白数据传送到 RAM 装置。

图 1 是关于本发明的一个实施例的记录装置的电路方框图；

图 2 是在该实施例中的游程扩展器的方框图；

图 3 是另一种流程扩展器的方框图；

图 4 表示在该实施例中根据游程压缩方法的游程代码表；

图 5 是本发明的另一个实施例的记录装置的电路方框图；

图 6-1 至图 6-3 示出了根据该实施例的数据和预测函数值表的状态组合；

图 7-1 和图 7-2（1）至 7-2（18）示出了在该实施例中根据反向预测函数处理的数据的重建状态；

图 8 图示了该实施例的反向预测处理器的结构；

图 9 是本发明另一个实施例的一种记录装置的电路方框图；

图 10 是在该实施例中的缩小处理器的方框图；

图 11-1 至图 11-4 用于解释减少位的转换方法；

图 12 是 2/3 缩小转换器的方框图；

图13是4/5缩小转换器的方框图；

图14是7/8缩小转换器的方框图；

图15是1/2缩小转换器的方框图；

图16是UNIT部件的方框图；

图17是(48×48)点阵打印字形的示例图；

图18是规范化字形的示例图；

图19是压缩字形的示例图；和

图20是本发明第四实施例的游程扩展器的电路结构的方框图。

现结合附图对本发明的实施例详细地描述。

[实施例1]

在象字形数据的具体数据的情况下，当考虑到宽的视野，可以说所有这些数据都具有规律性。当注意到一个字形，如果是双向代码，2^{n+1}种状态是以相对于一个象素(点)所过去的(n)个象素的相互组合的数而呈现的。然而，对所有的状态提供那样的逻辑实际是不可能的。因此，仅注意到一个象素的周围的象素。特别是，如果是可以与一般的图象数据相比较的字形数据，从黑点到白点以及从白点到黑点的变化点的数目是小的。因此，考虑了一种方法，借用它通过用黑点表述变化点，白数据部分的数量增加了，而且游程压缩的效率也增加了。另一方面，虽然从白点到白点以及从黑点到黑点的跃迁几率是很高的，从白点到黑点以及从里点到白点的跃迁几率是很低的。因此相邻于一个象素的多个象素的二元状态能够预测。通过将前面的概念应用于原来的字形数据，该数据变形为一种形式，其优点在于使用预测函数进行游程压缩。

由于在诸如字形数据的二进制数据中黑数据组与白数据组肯

定交替地出现，每一组用游程码来表示。从垂直方向的顶部顺序地检索字形，并且使用游程码（图4）压缩连续的白点。当搜索点到达字形的下边，一个右列的开头的点被认作连续地到达下边．因此它是被一维地编码的。至于游程码，在〞1〞以前的连续的0至22位（称之为操作数）的〞0〞与随后的具体的位串（称之为数据）（由操作数的长度所决定）组合实现编码的。就是说：假定先于〞1〞的连续的〞0〞的位数（等于操作数部分）等（n），那么，从预定的数据长度表中所得到的数值将被设定为随后的〞数据部分〞的位的长度。通过将相应于（n）的数据初始表的项目加到跟在〞1〞以后的m位数据部分的值（称为a）上所获得的值用于指示连续白点（白数据外）的数量。虽然白数据块和黑数据块必定交替出现，但由于必须判断字形的数据块首是白数据块还是黑数据块，所以假定表示数据块首是白数据块或黑数据块的1位的数据识别符存储在压缩数据的开头。具有被如前所述进行压缩的游程的字形数据存储在存储器部分。当转换来自主计算机输入数据的打印图形时，通过一个硬件执行游程扩展处理过程。

图1是根据本实施例记录装置控制系统结构的电路方框图。在图1中，参考数101表示在记录装置中执行运算和处理的CPU；ROM102用以存储CPU101的控制程序、产生字形的游程压缩数据以及用作处理的各种数据；RAM103暂存储根据发白主计算机的打印数据由记录该打印图象数据以及各种数据；数据接收机100用来执行数据接收机与诸如主计算机或类似的外部装置间的通讯控制，并用以接收主计算机发来的打印数据；打印头驱动器105用于将来自CPU101的一个控制信号提供给打印头驱动器，以便利用所述的记录装置将一

个图象记录到一个记录介质上，并且用于传送图象数据到记录头上和打印；CR驱动器106用于托架马达驱动器和一个托架马达，

送纸马达以及送纸马达驱动器的LF驱动器107是一个控制系统，用于将来自CPU101的控制信号送给每个驱动器以便引起记录头在水平和垂直方向上的运动，而该记录头是在记录介质上打印时所必要的。104表示游程扩展器，该扩展器用于将ROM102中存储的游程压缩字形重建成为一个打印位图形。

在本实施例中的一个控制过程现将参照图1进行描述。首先，由数据接收机100接收的数据暂存储在RAM103中。命令，图象数据以及字符代码由CPU101根据存储于ROM102中的程序进行分析。在字符代码的情形，先前已被游程压缩和存储的字形数据从ROM102中读出并传送给游程扩展器104。在游程扩展器中，字形的多个位产生出来并顺序存储于RAM103中。在当一行图象数据的转换被完成，或者从外部装置的主计算机输入一个打印命令的时刻，CR驱动器106被驱动并且从RAM103传送的打印数据传送到打印头驱动器105并且被打印。在完成打印后，LF驱动器107被驱动并且完成一串的时序。

在图2中的游程扩展器104的方框图将详细描述。图2示出了这样一种情况，即所述CPO的存取单元被设定为一个字（16位）并且在垂直方向上被压缩的压缩字形被扩展。对于游程扩展的要求通过一个控制标记FLG-CTRL从CPU中产生。根据图7-1以及图7-2（1）至图7-2（18）的游程码表在先前所形成的压缩字形通过适量的一个字传送到数据阅读器200。当所述的数据传送被确认，通过一个数字移位器202将所述数据逐位地顺序移位。数201表示移位计数器用

以计算移位次数的数量。当检测到移位处理的执行利用的仅仅是压缩字形的位数并且移位处理已经完成，压缩字形的一字结束标记FLG-COMP被设定。在一个字的下面的压缩字形被传送的时间点上，该压缩字形的一字结束标记被清除。首先，第一个数据的第一位被传送到标题鉴别器203，通过该鉴别器作出检查以看看随后的位是白色位还是黑色位。表明鉴别结果的信息被传送到一个操作数、数据和黑色数据的转换电路204。数据205表示一个白色数据的操作数分析器用以计算由数据移位器202移位的〝0〞的位数，并且在检测到〝1〞的时间点处通过计算值获得相应操作数长度的数据初始值和用以传送数据初始值给一个锁存电路206。操作数分析器205获得数据部分的位长度，该数据部分已由每个操作数长度设定，并且传送表示该位长的信息给数据分析器208。通过操作数分析器205所作操作数分析的结束被通知给转换电路204。该数据分析器208由转换电路204启动。该数据分析器208检测随在操作数之后的移位数据的〝1〞和〝0〞的位数，并且获得图7-1中以及图7-2(1)至7-2(18)中数据可表示范围内的数值和传送给一个数据锁存电路209。

通过数据分析器208对数据进行分析的结束顺序地通知转换开关电路204。在锁存电路206和209中锁存的值提供给一个加法器207并且使之相加。该加法结果传送给一锁存电路210作为白数据的位数。数据移位器202的数据移位处理一旦中断，白数据的字形转换过程就被启动。例如，现在假定操作数部分和数据部分的数据被设为000010100[最低有效位(LSB)←→最高有效位(MSB)]，操作数部分设为〝0001〞，从图4将可以明白，这时的数据初始值等于〝49〞，并且数据部分的位度等于4位。该过程直至现在是由操作分析器205

执行的。

由于数据部分是〃0100〃(LSB←→MSB)，该值等于〃2〃，由数据分析器208执行这一分析。在相加后，白数据位的长度等于〃51〃，锁存在锁存电路210中的白数据位的数由白数据位图形转换器212顺序地转换成为垂直位串。由一个字形发生器214形成字形矩阵。在垂直方向上形成的16位的字形传送给锁存电路216并将它的信息作为控制信号加给CPU(规范字形一字完成标记FLG-NORM)以及该字形数据存储在RAM中。

一个位计数器215计数所转换的位数，判断结束，通知该结束给数据移位器202，允许移位过程重新启动以及启动黑数据分析器211。黑数据分析器211检测从数据移位器202移出的位的〃1〃或〃0〃，以及设置仅仅[(〃0〃位的数)+1]的位给黑色位，并且启动黑色数据位图形转换器213。例如，在连续四个黑点的情况下，他们表示一个操作数，并且随在他们后面的数据表示为〃0001〃。

上述处理的执行的总共次数仅是由移位计数器201移位数据的位数。当检测到移位处理已经完成，从ROM中读压缩数据的处理再继续进行。上述过程重复到一个字的规范字形已形成。有关结束的判断是由CPU通过对打印字形的字的计数来完成的。

在实施例1中，在垂直方向上完成16位的转换这一时间点，控制信号(规范字形一字完成标记)传输给CPU。CPU通过该控制信号读出该转换的位字形并且存储于RAM中。然而，如图3所示的游程扩展器中，提供DMA控制器220以代替数据阅读器200。在沿垂直方向完成16位的转换以后，经字形传送到RAM(DMA传送)，而不需要通过CPU传输。用一个传送结束鉴别器217判定一个字符的转换的结束

以便计算一个字符的数据，借此停止DMA控制的启动。DMA控制的停止被确认以及一个字符的转换的结束通过一个标记通知CPU。由于这个，处理速度能进一步提高。在有多种字形尺寸存在的情况下，提供了具有一个开关的字符尺寸设定器218，在传送一结束鉴别器217中的一个字符的数据计数器制成为可变的，借此也使得该装置对不同尺寸的字形是有效的。在寄存来自主机的诸如下装字符或类似字符的外部字符的情况下，通过将游程压缩处理过程作为一个程序而存入一个内部RAM，可以使一个外部字符寄存区域极大地减少，以至小于通常寄存区域的规模。那样一种结构对减小RAM的容量也是有帮助的。

根据前述的实施例1，在记录装置中的字形数据的信息数量能够减少，而且成本也能降低。此外，用硬件来实行游程压缩数据的扩展处理，制作字形数据的处理速度能够提高。软件的装入能够相当地减少，整个记录装置的信息通过量能够改善。

[实施例2]

图5是根据实施例2的记录装置的控制系统结构的电路方框图。100至107是与图1所示的相同部件单元，而且实施例2除了存在反向预测处理器108外和实施例1是相同的。

在图5中，101表示CPU，它在记录装置中执行运算和处理；ROM102用于存储CPU101所使用的控制程序、形成字形的游程压缩数据以及用于处理的各种数据；RAM103暂时存储图象数据以及根据来自主机的打印数据将由打印头打印的各种数据；数据接收机100用来执行数据接收机和象主计算机或类似的外部装置间的通讯控制以及用来取从主计算机传送的打印数据；控制信号来自CPU101并

加于打印头驱动器105，以便通过记录装置在记录介质上记录图象，并且它传送图象数据给记录头和打印；和用于托架马达驱动和一托架马达的CR驱动器106。

送纸马达驱动器的LF驱动器107以及送纸马达是一个控制系统，用于从CPU101提供一个控制信号给每个驱动器和用于引起向记录介质上打印时所必要的记录头在水平方向和在垂直方向的运动。104表示游程扩展器用以将存储在ROM102中的游程压缩字形重建成打印位图形。108表示反向预测处理器用于从游程扩展器104扩展的据重建原来的字形。

按照前述过的类似方法并参照图5现在将描述该实施例中的控制过程。首先，由数据接收机100所接收数据一旦被存储在RAM103中，CPU101就根据存储在ROM102中的程序分析命令、图象数据和字符代码。如果是字符代码，已经由前面述过的预测函数做过数据转换、以及游程压缩和存储的字形数据从ROM102中读出并传送到游程扩展器104中。由游程扩展器产生字形的多个位并且顺序存储在RAM103中。该存储的数据传送给反向预测处理器108并形成一个打印字形图形。在当完成了一行的图象数据的转换或当从作为外部设备的主计算机输入打印命令的时间点，CR驱动器106被驱动。打印数据从RAM103传送到头驱动器105并且被打印出来。在完成打印后，LF驱动器107被驱动并且完成了一系列的运算顺序。

根据图6-1中的预测功能值表现在将要做出解释。

在注意到图6-2中示出的字形中存在的一个点 (d) 的上面位置 (b)，左边位置 (c) 和倾斜的位置 (a) 的那些点的情况下，某一个点被称之为予测函数，在该点内，出现 (d) 的概率相对于三点 (a、b、c、

)组合而言要高。对于(a至c)这些点预测函数的数值可以和实际的点(d)的数值相比较。当他们是一致的,(d)的值由〃0〃(白)代替,当他们有差别时,(d)由〃1〃(黑)替代。也就是说高的转变概率的图形由白代替。由于这个,大部分的点由白替代,而且游程压缩的压缩效率提高了。

图6-3中示出的矩阵由下列预测函数方程表示。

Func (A, B, C) = ($\bar{A}$&B) 或 ($\bar{A}$&C) 或 (B&$\acute{C}$)

……方程1

如前所述,通过预测函数转换原来的数据,该转换成的数据被进行游程压缩,该压缩的数据存储在记录装置的存储器中。当来自象主计算机或类似的外部设备的字符数据被转换成为字体时,首先根据存储在存储器中的压缩数据执行游程扩展,并且构成了一个反向函数处理的逻辑电路,以用于已经进行游程扩展的数据。

再建处理的反向预测函数的逻辑方程如下(参考图7-1)

Y=Dxor ($\bar{A}$*B+$\bar{A}$*C+B*C) ……方程2

Y=输出数据

D=未处理的数据

图7-2(1)至图7-2(18)示出了再建处理的实例。

图7-2(1)涉及的数据是对原始字形数据执行了予测函数处理以后所形成的数据。该数据是根据方程(2)从存在于上述数据左上边的点顺序重建的。在图7-2(2)至7-2(17)中,一个位于左和上边的外周边的一个点实际上被认为是白色(0)。由黑框包围的四个点中,位于相关点〃D〃上面〃B〃位置、左边位置〃C〃和斜对着的位置〃A〃的三个点被取出。点〃D〃的值被检查。如果D=〃0〃,这意味着D的值

和预测函数值是相符的，如果D="1"，这意味着D的值与预测函数值是不同的。

当考虑到图7-2(2)的矩阵时，就由黑框包围的四个点而言，相应于"D"的点被设定为黑色"1"，并且和预测函数值不同。因此，根据图2的预测函数值表，将认为该点设定为黑"1"。在图7-2(3)中的处理现在将进行。在这个例子中，在图7-2(2)中产生的点被用作存在于右上位的点。用和7-2(2)类似的方法，当注意该点"D"时，它等于白"0"，并且和预测函数值一致。在这种情形下，根据图2的预测函数值表将可以理介，图7-2(3)的矩阵对应着"3"的图形，并且该点等于黑"1"，借助于连序重复上面的处理过程，图7-2(18)的图形重新构成。通过硬件，满足前面逻辑方程的逻辑构成每一位，而且实现了所希望的字体数据的格式。

图8是反向预测处理器108的硬件结构的示图。在前面的逻辑中，每16位执行转换运算。前一行所转换的数据传送到AB15至AB0，而没有转换的数据传送到D_{15}至D_0。与数据传送完成同步，反向预测函数处理逻辑开始工作并且对应这些位的数据输出给Y_{15}至Y_0。通过读出Y_{15}至Y_0，可以得到执行过反向预算函数处理的数据。该最低有效位被锁存以用于下面点串的最高有效位的处理。在完成读出操作后，该输入部分被清零。

虽然在前面实施例中每16位执行一次反向预测函数处理，通过在与每一位都相关的字形尺寸的垂直方向上安装相应一个点串的数的逻辑电路，就不需要用软件清零输入部分。处理速度能够提高，并且不需要传输给CPU就能转换的数据也能够传送到RAM。在寄存象来自主计算机的下装字符或类似的外部字符的情况下，通过存储

游程压缩处理和预测函数处理作为内部ROM的程序，外部字符寄存区能明显地减小到比通常寄存区要小。本发明也能有助于减小RAM的容量。

根据前述的实施例2，通过用预测函数转换原字形数据，黑点的大部分能由白点代替，而且游程压缩的压缩效率能大大提高。通过用硬件执行反向预测函数处理，CPU(软件)的装入可以减少，字形数据形成的处理速度也能提高，整个记录装置的通量也能改进。

(实施例3)

如果在垂直方向上减少数据，在减少以前位于数据中预定位置的点按每个减少率(图11-1至图11-4)变的稀疏。在图11-1中，数据按1/2的减少率每隔一点稀疏掉一个。在图11-2中，数据按2/3的减少率每三个点稀疏掉一个点。在图11-3中数据按4/5的减少率每五个点稀疏掉一个点。在图11-4中数据按7/8的减少率每八个点稀疏掉一个点。建立一个逻辑以致可象图11-1至11-4那样转换数据，从而实现了高的处理速度。

图9示出了根据本实施例的记录装置的控制系统的结构的电的方框图。在图9中，901表示CPU用以执行记录装置中的运算和处理；ROM902存储为CPU901用的控制程序和字体；RAM903用以暂存储由打印头要打印的图象数据和各种类的数据，这种打印是以从主计算机传来的打印数据为基础的，数据接收机900用以执行数据接收机与象主计算机或类似的外部装置间的通讯控制以及得到从主计算机传来的打印数据。头驱动器905用于将来自CPU901的控制信号提供给头驱动器，以便利用记录设备在记录介质上记录图象，并且用于传送图象数据给记录头和打印；以及托架马达驱动器和托架马达的

CR驱动器906。送纸马达驱动器和送纸马达的LF驱动器907是从CPU901给每个驱动器提供控制信号的控制系统，并且使得打印中所必要的记录头沿水平方向和垂直方向的运动。904表示一个缩小处理器，用于根据在CPU901中设定的缩小率在垂直方向执行位转换。

用类似前述的方式结合图9将描述本发明的控制程序，首先由接收机900接收的数据一旦存储在RAM903中，就由CPU901根据存储在ROM902中的程序分析命令，图象数据和字符代码。如果是字符代码，先前已经存储在ROM902中的字形数据顺序读出，并且一个具位图形存储在RAM903中。存储的位模式传送到缩小处理器904，904是根据由CPOU设定的格式缩小比率工作的。每个希望的位图形形成并再次存贮在RAM903中，同时，当一生的图象数据转换完成或者当从作为外部装置的主计算机来的打印命令输入了，该CR驱动器906被驱动，而且来自RAM903的打印数据传送给头驱动器905，并且被打印。当该数据传送到头驱动器905时，在横向方向上的数据也同时减少。这种在横向方向的数据减少是在一个字节单元基础或在一个字单元基础上通过稀疏掉数据的方法实现的。在完成打印以后，LF驱动器907被驱动而且一串运算序列完成了。

图10示出了缩小处理器904的硬件结构。用前面的逻辑，在垂直方向上的64点转换前的数据首先由CPU以每个字四次的方式写入，而且该数据分别传送到

BO×OD(15:0)，　BO×ID(15:0)，　BO×2D(15:0)和

BO×3D(15:0)

而且传送到转换器910到913，用来分别按图11中示出的缩小比率1/2，2/3，4/5和7/8沿垂直方向将数据减少。转换器的输出传送到

16位的寄存器REG，而且通过从CPU读出输出Y_0到Y_{15}，可以获得减少的数据。

图12至图16示出了转换器910至913内部电路结构。图12示出了转换器910执行1/2的数据减少。图13示出转换器911执行2/3的数据的减少。图14示出了转换器912执行的4/5的数据的减少。图15示出了转换器913执行的7/8的数据的减少。图16示出了图13至15中的UNIT的内部结构。在该图中，ORCLX表示了一个开关以选择简化稀疏位的模式和在一点和邻近点间得到或(OR)的模式中的一种。CS(2:0)表示地址解码。当由CPU读出减少的数据时，在一个字单元基础上CS的值读出四次。CMP2(15,8)，CMP3(15,0)，CMP5(15,0)和ZMP8(15,0)表示转换后的数据。

在实施例3中，当缩小时，通过稀疏掉多个点来形成数据。然而，在产生的数据的质量变成问题的情形，通过提供一个开关以便从在将要稀疏的数据和相邻数据间得到或(OR)的模式以及简化的稀疏的模式中选择一种，并且通过组合那个电路在该装置中，一个缩小的位图形能够产生。正本字符(上标字符，下标字符)也可以用那种电路形式，而且依赖于字符种类的软件的处理也能减少。

如前述的实施例3，当执行了格式的缩小时，通过用硬件实现垂直方向的位转换，软件的装入可以减少，数据的形成的处理速度能提高，整个记录装置的信息通量能够改善。

(实施例四)

图17示出了由实施例1的记录装置打印(48×48点)的字形的一个例子(亚，读"a"的中国字)。图18是一个规范的字形，它是通过用预测函数规范图17的打印字形得到的。图19中的16进制的符号表

示了通过游程压缩了图18的规范字形而获得的压缩的字形，并且压缩的字形以这样一种格式存储在ROM中。在记录装置中，在ROM中存储的图10的压缩字形通过游程扩展器扩展成图18的规范字形，而且，随后，通过反向预测函数再建为图17的原来打印的字形，借此，获得打印数据。

当以图19所示格式存储于ROM中的压缩字形用图6中的游程扩展器扩展为图18中的规范字形，并且当使用预测函数对黑位的缩小比率是高的时，连续的白位的长度变得长了，所以，白位连续的部分就出现在多个字的上方。当考虑从点A处的黑位到点B处的黑位（图18中）的一部分作为例子时，357个白点连续。当他们在一个字单元基础上进行划分时，21个字变为仅包含白位的字，也就是白字（数据S0000的字）。在图18的情形中，白字数等于103，它几乎类似所有字数（144）的3/4。

在图6-1至图6-3中的游程扩展器中，即使在白字连续的情形下，在逐位移位的同时，也能够形成规范字型的一个字。因此，在完成一个字以前，CPU不检测该字是不是白字。在其中存在很多白字的规范字形的情况下，将占据很长时间用于游程扩展。

在这个实施例中，由于根据白位数计算白位的字（白字的数）数的白字记数器已加入到图6-1至图6-3的游程扩展器中，由CPU可以读连续白字的长度。因此，假如白字出现，游程扩展器的硬件减少相应移位所需时间的CPU的等待时间。存在白字的规范字形的游程扩展处理的处理速度提高了。整个记录装置的信息通量也能改善。

除去仅仅是字计数器不同外，根据本实施例的游程扩展器的方框图几乎类似于图6-1至图6-3。因此现在将解释不同点。其他的

结构类似实施例1和2。

根据这个实施例，图20示出了游程扩展器的电路结构。

在图20中，由200至216表示的结构部分表明了类似图6-1至图6-3的功能方框，这里省去描述。在图20中，除了由CPU写入的游程扩展请求（控制标记）之外，虽然在图61至图6-3中没有表示，但是压缩数据显示出来了，而且来自位计数器215的规范字形一字结束标记NORM以及来自移位计数器201的压缩字形一字结标记FLG-COMP作为由CPU读出的信息表示出来了。230表示的白字计数器作为功能块加入该实施例。当来自于白数据锁存电路210的白位数小于在位计数器215中形成期间一个字的剩余的位数时，实际上该白字记数器230传送白位的数给白数据位图形转换器212，并且产生〃0〃作为由CPU去读的白字的字数。当白位的数等于或大于剩余位的数时，剩余位的数传送到白数据位图形转换器212。产生的商数（在二进制记数法中4位或更多的位的高位）作为由CPU读的白字的数，而该商数是通过从白位的数减去剩下的位数除以16而得。同时，该余数被锁存和保持以便产生下一个字。该余数在规范字形形成并由CPU读出后传送到白数据位图形转换器。

现在将描述本实施例中的CPU的工作过程。

在游程扩展处理由控制标记启动，并且压缩字形被写入后，压缩字形一字结束标记和规范字形一字完成标记被监视。当压缩字形一字结束标记被设定时，该一字的压缩字形被写入。当规范字一字完成标记被设定时，规范字形的一个字和白字的字数被读出，并且规范字形传送到RAM。当白字的字数等于零，处理程序返回到标记监视器。当白字的数不等于零，（仅和白字的数一样多的数的），

数据 0000 (仅包含白位的字的表示) 传送到RAM。

图18中的点A和B现作为例子来描述。当知道白位的数等于357, 白字计数器230将白位的数 (即357) 与剩余的位的数 (在这种情形是15) 相比较。由于 (在这种情形中的) 白位数等于或者大于剩余位的数, 位 (15) 的数传送到白数据位图形转换器212, 借此, 在形成期间完成规范字形的一个字。该商 (357-15=324; 324/16=21并且余数等于6) 的获得是通过从白位的数中减去剩余位的数除以16, 它随后设置为白字的数。该余数6被锁存并保持以用于下一个规范字形。该CPU检测规范字形一字完成标记已经设定, 读取该规范字形, 并且将其传送到RAM。随后, CPU读白字的数并以对应那个商值的次数来, 传送数据S0000, 即21次。因此, 该CPU就不需要等待白字形成21次。游程扩展的处理速度改善了。当规范字读出时, 游程扩展器清除规范字形一字完成标记并且同时传送已经锁存以用于规范字形下一个字的白位的数 (6), 到白位图形转换器。该处理程序前进到下一个字的形成。

在该实施例中, 虽然该CPU存取访问一个字 (16位) 单元基础, 但在存取一字节 (8位) 单元基础或长的字 (32位) 单元基础的情况下, 通过分别设置数据阅读器200和锁存电路216和移位计数器201和位计数器215的计数值为8位和32位, 该游程扩展器能够使用。

除了上面的实施例, 本发明也能应用到下面的实施例中。

1) 在实施例4中, CPU将压缩的字形写入到游程扩展器, 而且在一个字 (16位) 的转换被完成的时间点上, 检测到规范字形一字完成标记, 并且读出和在RAM中存储产生的规范字形。另一方面, 通过提供DMA区域用以在完成一字的转换后直接传送的数据到RAM而不需

要通过CPU传输，提供一个字符的数据的数据计数器（在48×48点的情形下计算高达144个字），和提供一个电路用以产生和白字的数一样多的数的数据$0000，判断一个字符转换的结束也可以在游程扩展器中执行。由于这个，能够进一步提高处理的速度。

2）另外，一个用以设定RAM的起始地址的电路加到DMA部分。在对应一行的数的ROM中的压缩字形存储起地址已存储在该RAM中，而且完成对应一行的数的游程扩展过程以后规范字形从CPU存储在该RAM中。所以一行的所有游程扩展过程可以由硬件执行。由于CPU的装入（软件）明显地减少，整个记录装置的信息通量能进一步改善。

3）还有，通过增加一个电路用以传送数据给反向预测处理器和在规范字形的一列（3字=48位，第四实施例中）数据的时间点将数据转换成打印字形，使得在记录装置的工作时序中的行程扩展以及由反向预测函数完成转换的所有部分，例如数据接收，字符代码的分析，游程扩展，由反向预测函数完成的转换，以及打印操作能由硬件来完成。因此，整个记录装置的信息通量大大地提高。

4）虽然在实施例4中字形尺寸设定为48×48点，但在有多种字形尺寸存在的情况下，通过新提供一个选择器，借此能够使移位计数器201的计数值和位计数器215的计数值成为可变的，游程扩展器也能用于不同的字形尺寸。

如前所述，根据实施例4，在仅包含白位的字（白字）的情况下，规范字形的游程扩展的处理速度可能提高，而且整个记录装置的通量能得到改善。

根据前面所述的本发明，数据形成的处理速度提高了，而且整个装置的信息的通量也能改善。

虽然这些实施例只使用记录装置加以叙述，但。图象数据也能显示在例如CRT，液晶显示装置或类似的显示装置上。

本发明能应用于系统，该系统由多个装置构成，也能应用在包含一个装置的装置。我们显然知道，本发明能应用这样的情形，即通过提供程序给系统或装置来实现本发明。

图 1

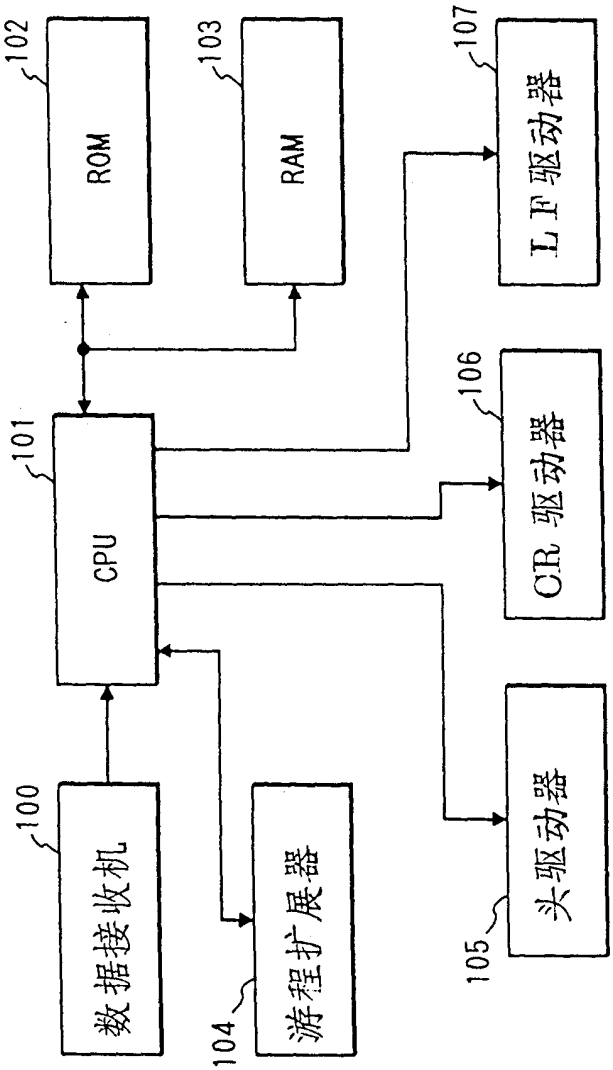

102
ROM
103
RAM
107
L P 驱动器
101
CPU
106
CR 驱动器
100
数据接收机
104
游程扩展器
105
头驱动器

图 2

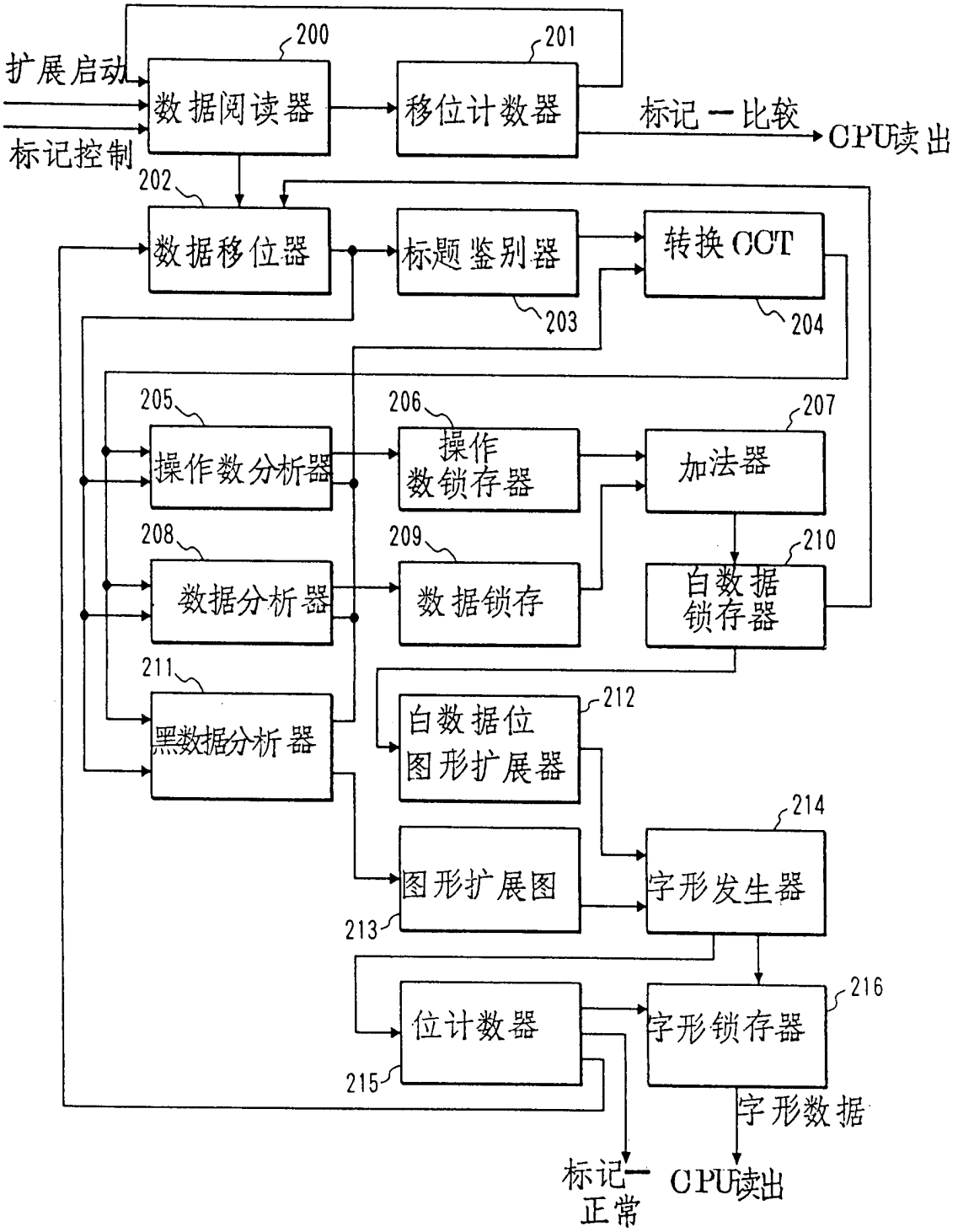

扩展启动
标记控制
200
数据阅读器
201
移位计数器
标记一比较
CPU读出
202
数据移位器
标题鉴别器
203
转换CCT
204
205
操作数分析器
206
操作数锁存器
207
加法器
208
数据分析器
209
数据锁存
210
白数据锁存器
211
黑数据分析器
212
白数据位图形扩展器
214
图形扩展图
213
字形发生器
位计数器
215
字形锁存器
216
字形数据
标记正常
CPU读出

图 3

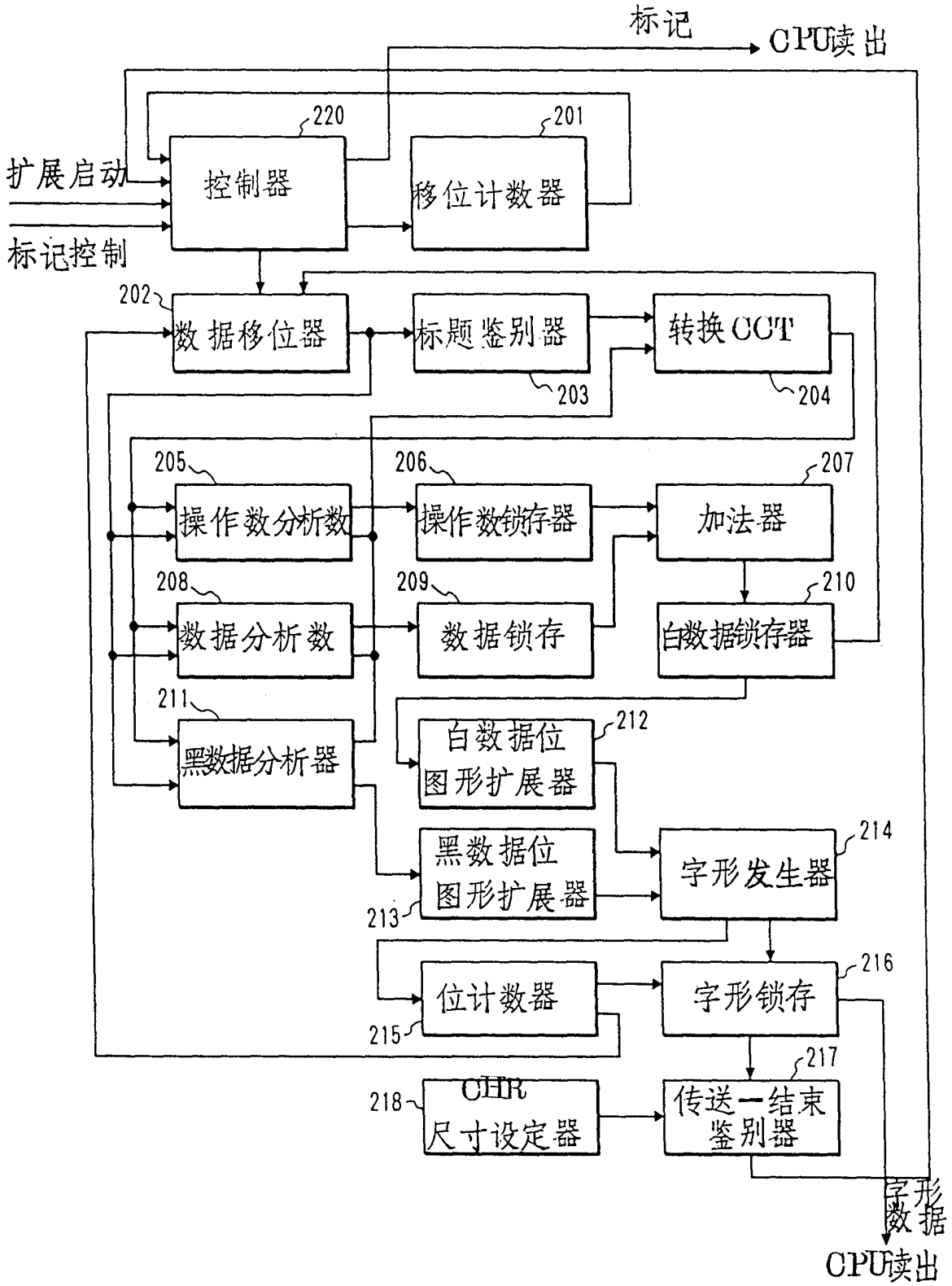

标记
CPU读出
220
201
扩展启动
控制器
移位计数器
标记控制
202
数据移位器
标题鉴别器
转换CCT
203
204
205
206
207
操作数分析数
操作数锁存器
加法器
208
209
210
数据分析数
数据锁存
白数据锁存器
211
212
黑数据分析器
白数据位
图形扩展器
214
黑数据位
图形扩展器
字形发生器
213
216
位计数器
字形锁存
215
217
218
CHR
尺寸设定器
传送一结束
鉴别器
字形
数据
CPU读出

图 4

操作数长度	操作数	数据部分	数据长度	数据范围	数据初始值
0	LSB← →MSB	1***	3	1-8	1
1		01***	3	9-16	9
2		001****	4	17-32	17
3		0001****	4	33-48	33
4		00001****	4	49-64	49
5		000001*****	5	65-96	65
6		0000001*****	5	97-128	97
7		00000001*****	5	129-160	129
8		000000001******	6	161-224	161
9		0000000001******	6	225-228	225
10		00000000001******	6	289-352	289
11		000000000001*******	7	353-480	353
12		0000000000001*******	7	481-608	481
13		00000000000001*******	7	609-736	609
14		000000000000001********	8	737-992	737
15		0000000000000001********	8	993-1248	993
16		00000000000000001********	8	1249-1504	1249
17		000000000000000001********	8	1505-1760	1505
18		0000000000000000001********	8	1761-2016	1761
19		00000000000000000001********	8	2017-2272	2017
20		000000000000000000001********	8	2273-2528	2273
21		0000000000000000000001*********	9	2529-3040	2529
22		00000000000000000000001*********	9	3041-3551	3041

图 5

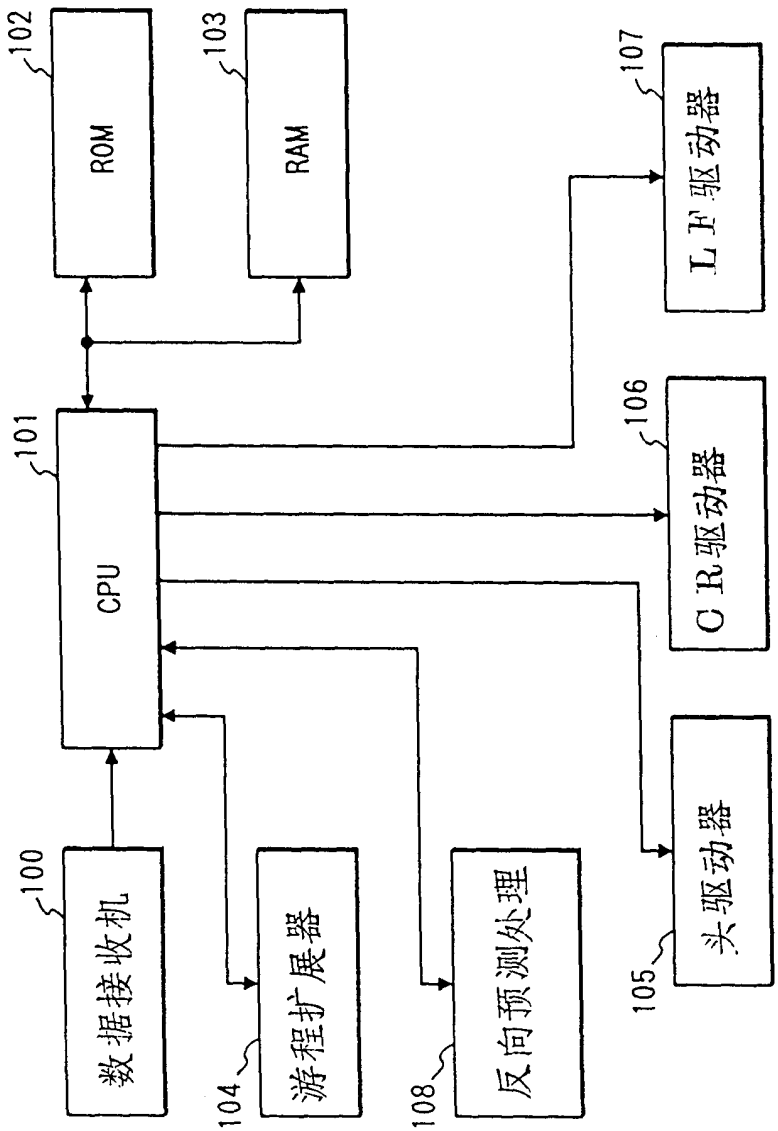

102
ROM
103
RAM
107
L P 驱动器
101
CPU
106
C R 驱动器
100
数据接收机
104
游程扩展器
108
反向预测处理
105
头驱动器

图 6－1

	a	b	c	预测值
(1)	0	0	0	0
(2)	0	0	1	1
(3)	0	1	0	1
(4)	1	0	0	0
(5)	0	1	1	1
(6)	1	0	1	0
(7)	1	1	0	0
(8)	1	1	1	1

图 6－2

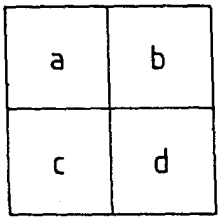
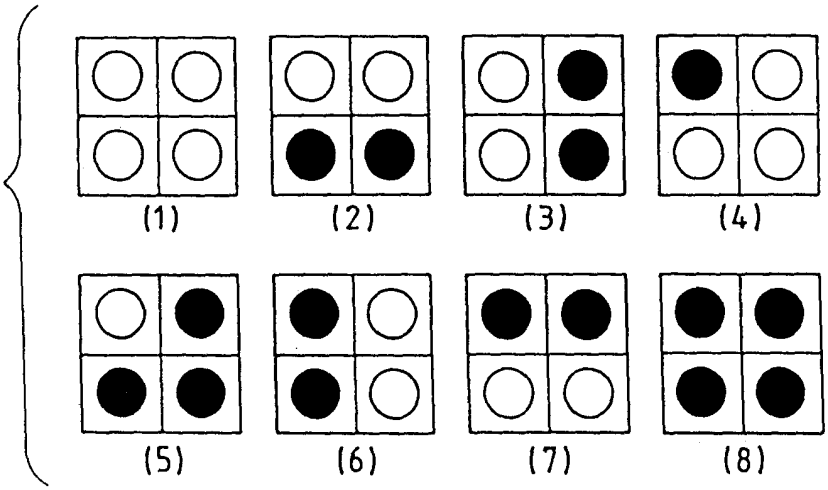


图 6－3



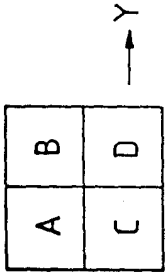

A
B
C
D
Y

图7－1

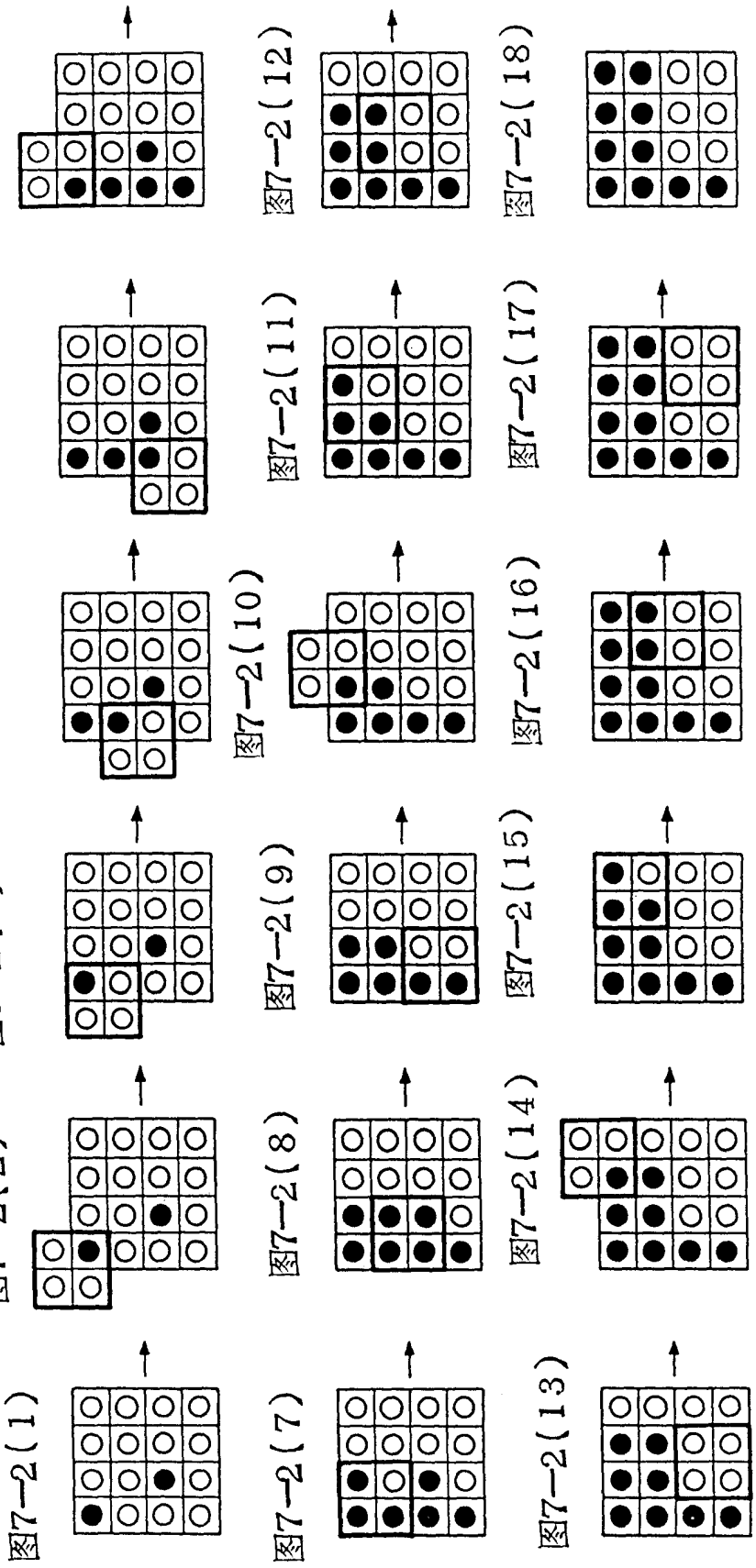


图7－2(1)
图7－2(2)
图7－2(3)
图7－2(4)
图7－2(5)
图7－2(6)
图7－2(7)
图7－2(8)
图7－2(9)
图7－2(10)
图7－2(11)
图7－2(12)
图7－2(13)
图7－2(14)
图7－2(15)
图7－2(16)
图7－2(17)
图7－2(18)

图 8

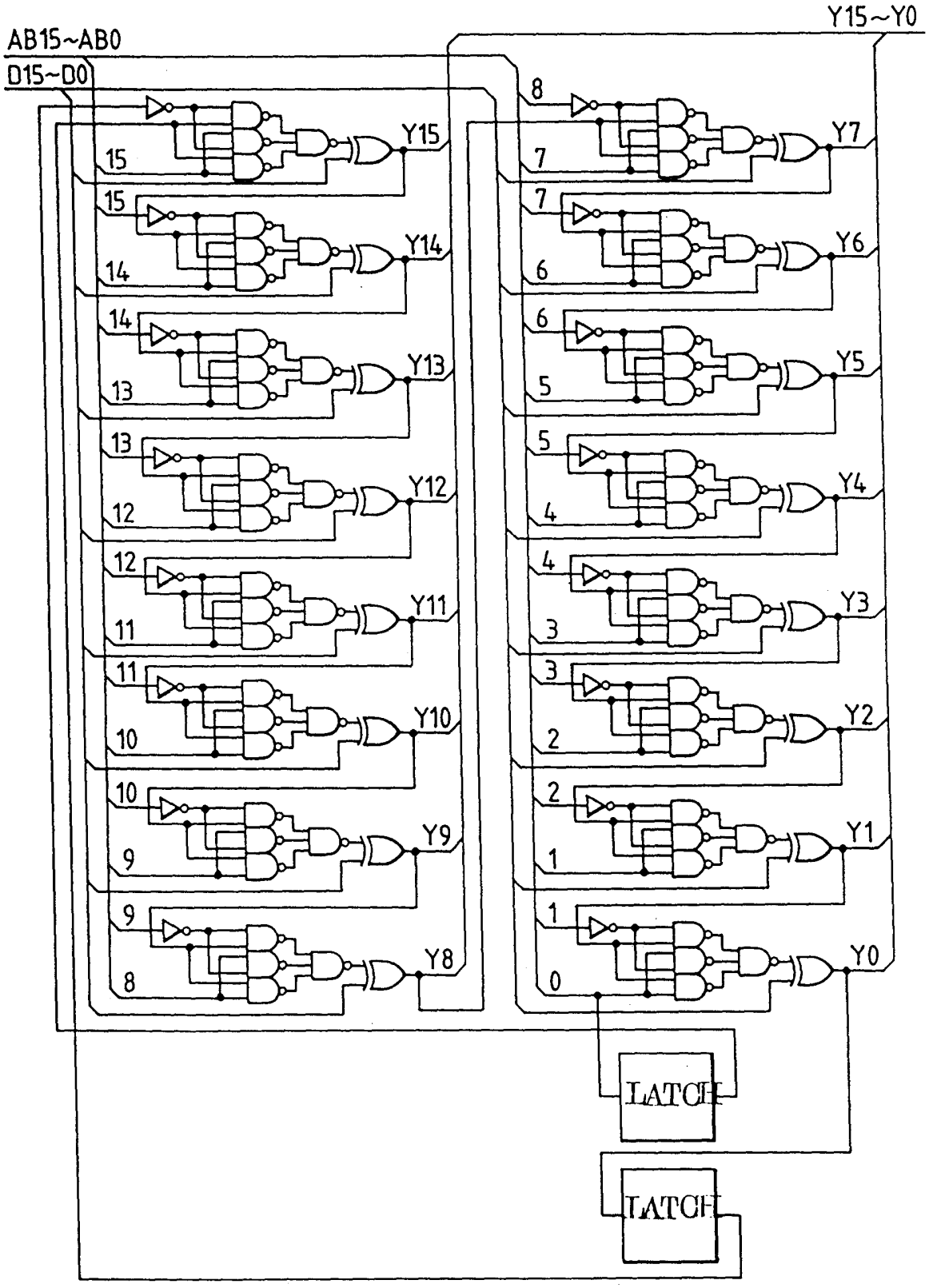

AB15~AB0
D15~D0
Y15~Y0
Y15
Y14
Y13
Y12
Y11
Y10
Y9
Y8
Y7
Y6
Y5
Y4
Y3
Y2
Y1
Y0
LATCH
LATCH

图 9

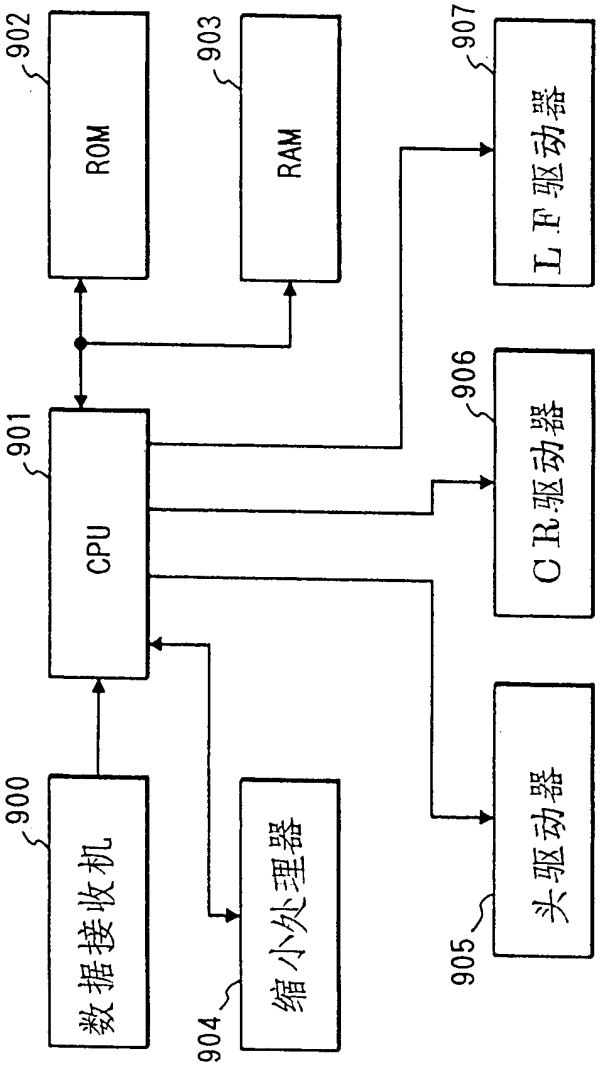

902
ROM
903
RAM
907
L F 驱动器
901
CPU
906
C R 驱动器
900
数据接收机
904
缩小处理器
905
头驱动器

图 1 0

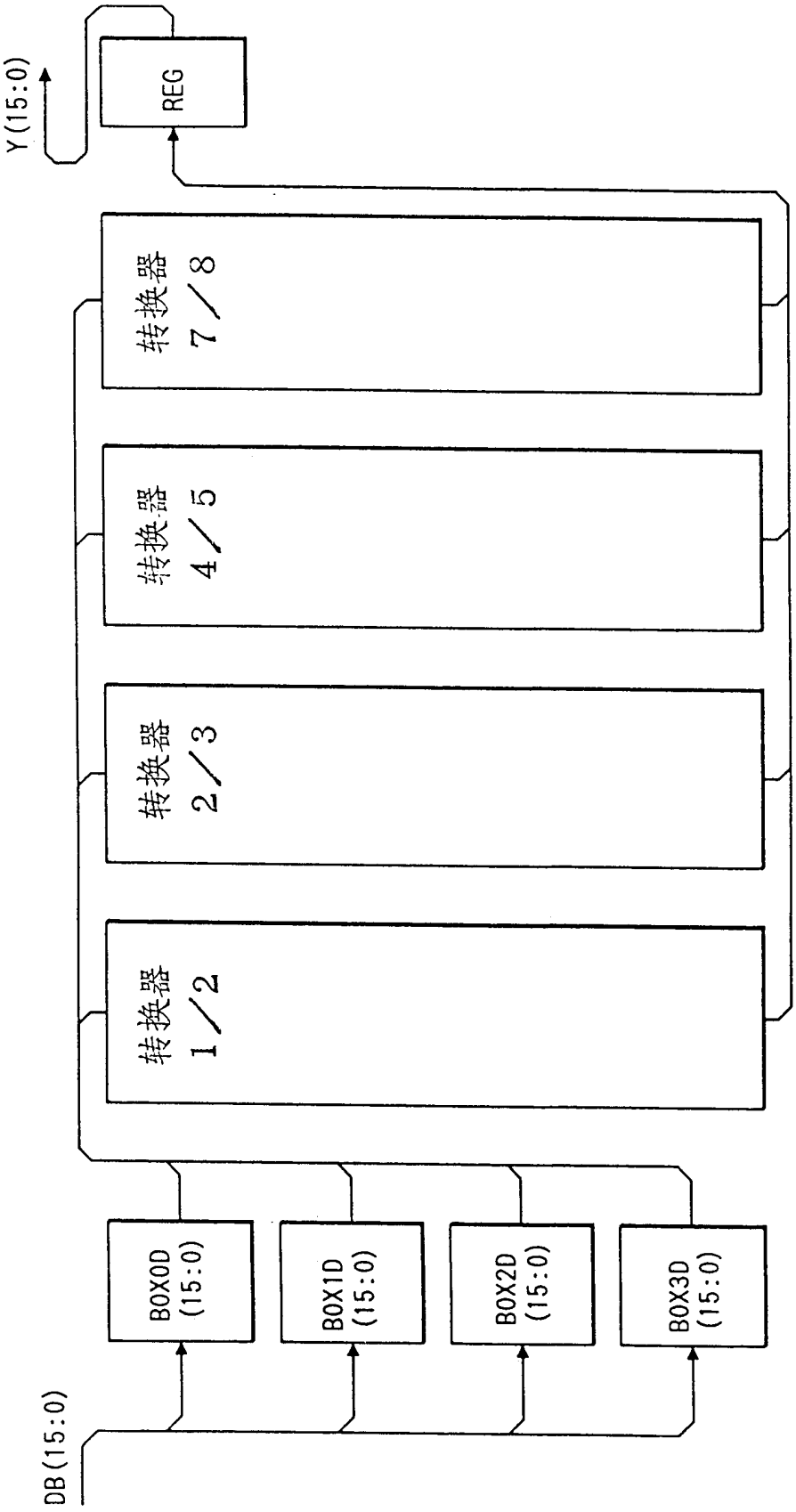

Y(15:0)
REG
转换器 7／8
转换器 4／5
转换器 2／3
转换器 1／2
BOX0D (15:0)
BOX1D (15:0)
BOX2D (15:0)
BOX3D (15:0)
DB(15:0)

图 1 1 — 4

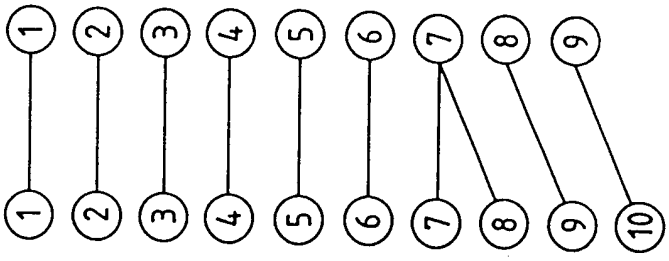


图 1 1 — 3

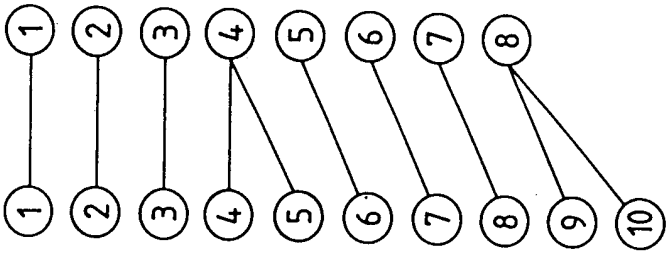


图 1 1 — 2

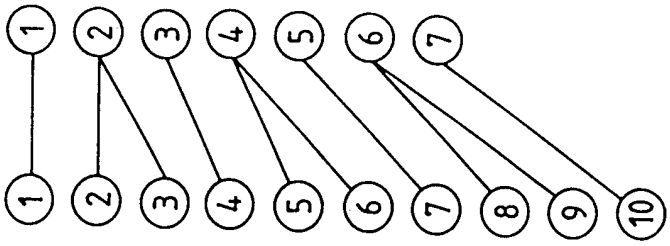


图 1 1 — 1

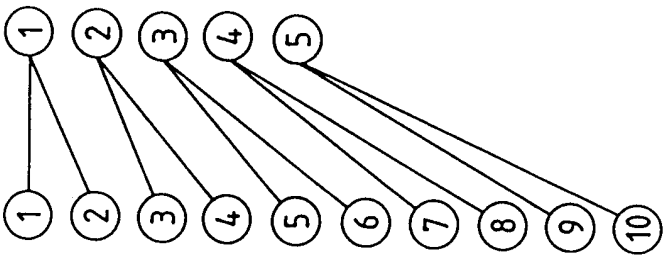


图 12

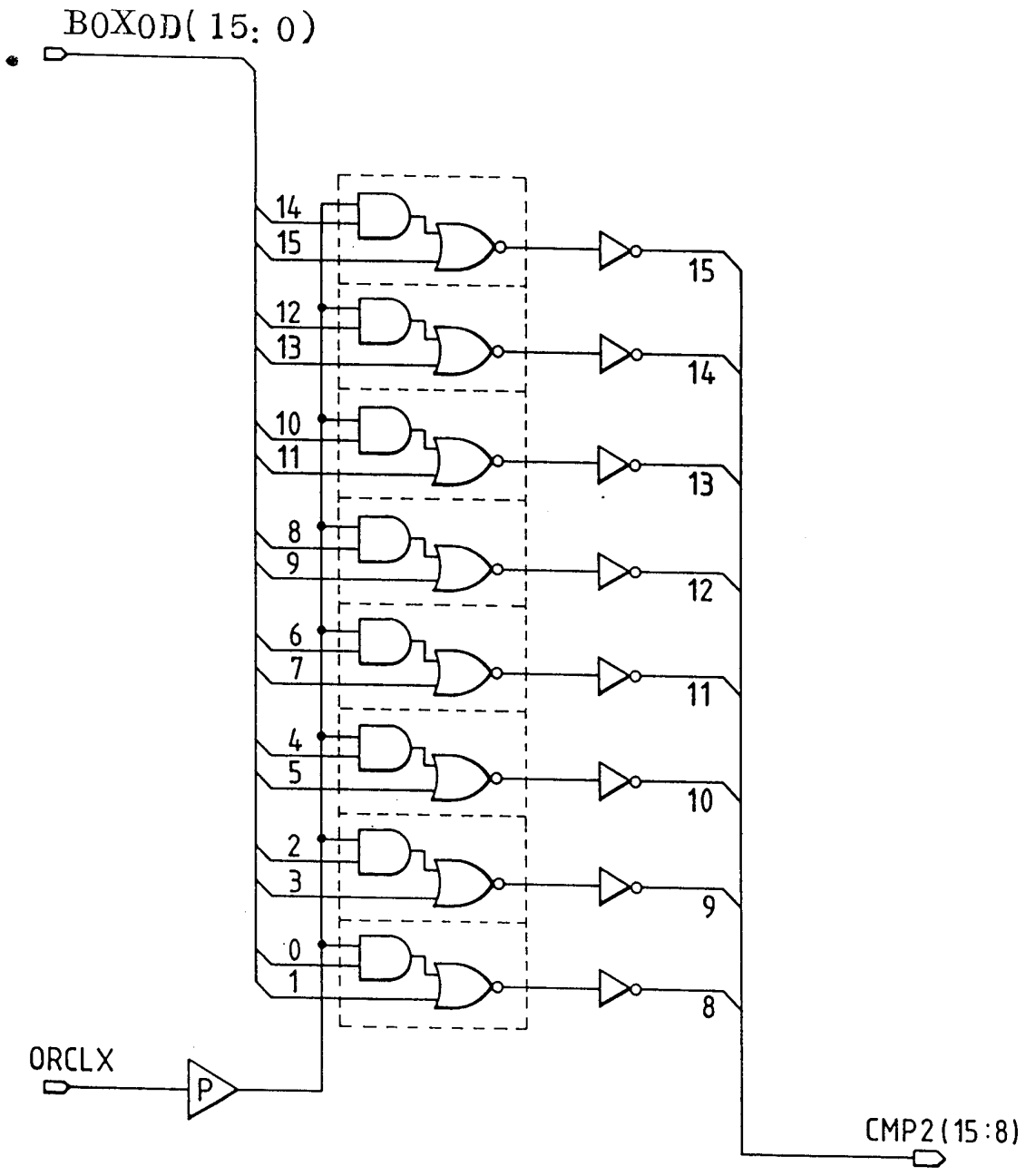

BOXOD(15:0)
14
15
12
13
10
11
8
9
6
7
4
5
2
3
0
1
15
14
13
12
11
10
9
8
ORCLX
P
CMP2(15:8)

图 1 3

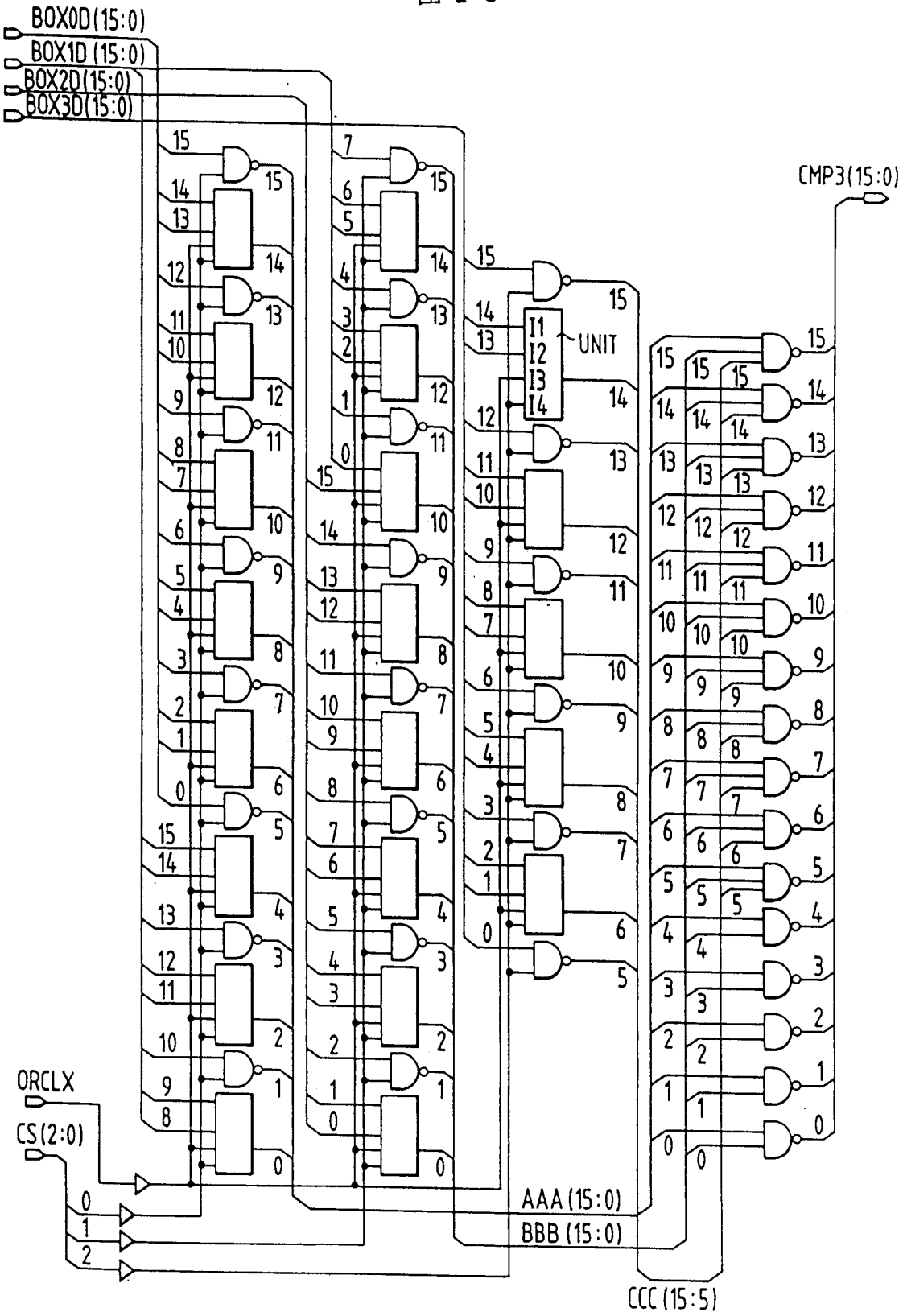

BOX0D(15:0)
BOX1D (15:0)
BOX2D(15:0)
BOX3D(15:0)
CMP3(15:0)
UNIT
I1
I2
I3
I4
ORCLX
CS(2:0)
AAA(15:0)
BBB (15:0)
CCC (15:5)

图 1 4

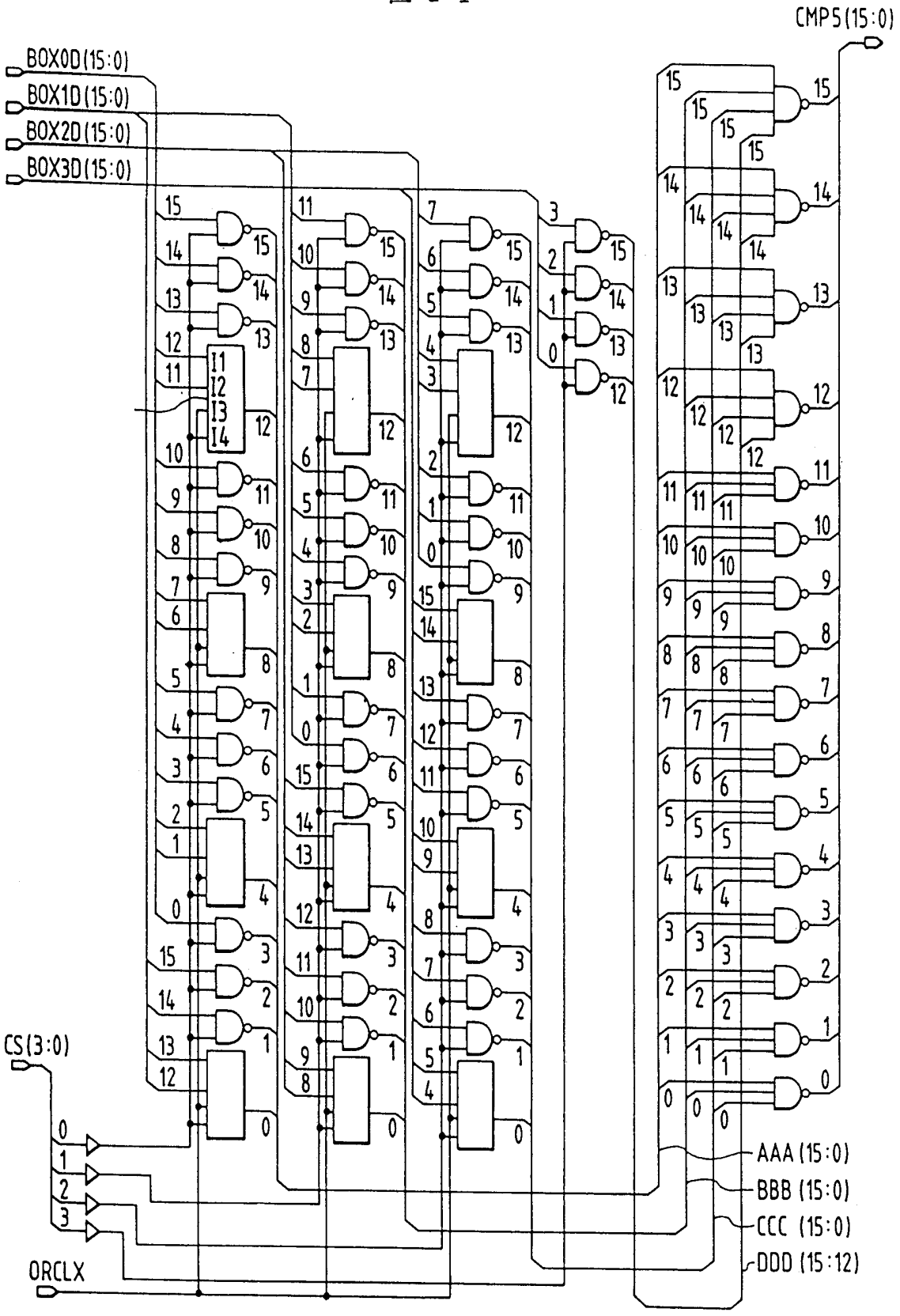

CMP5(15:0)
BOX0D(15:0)
BOX1D(15:0)
BOX2D(15:0)
BOX3D(15:0)
CS(3:0)
ORCLX
AAA(15:0)
BBB(15:0)
CCC(15:0)
DDD(15:12)

图 1 5

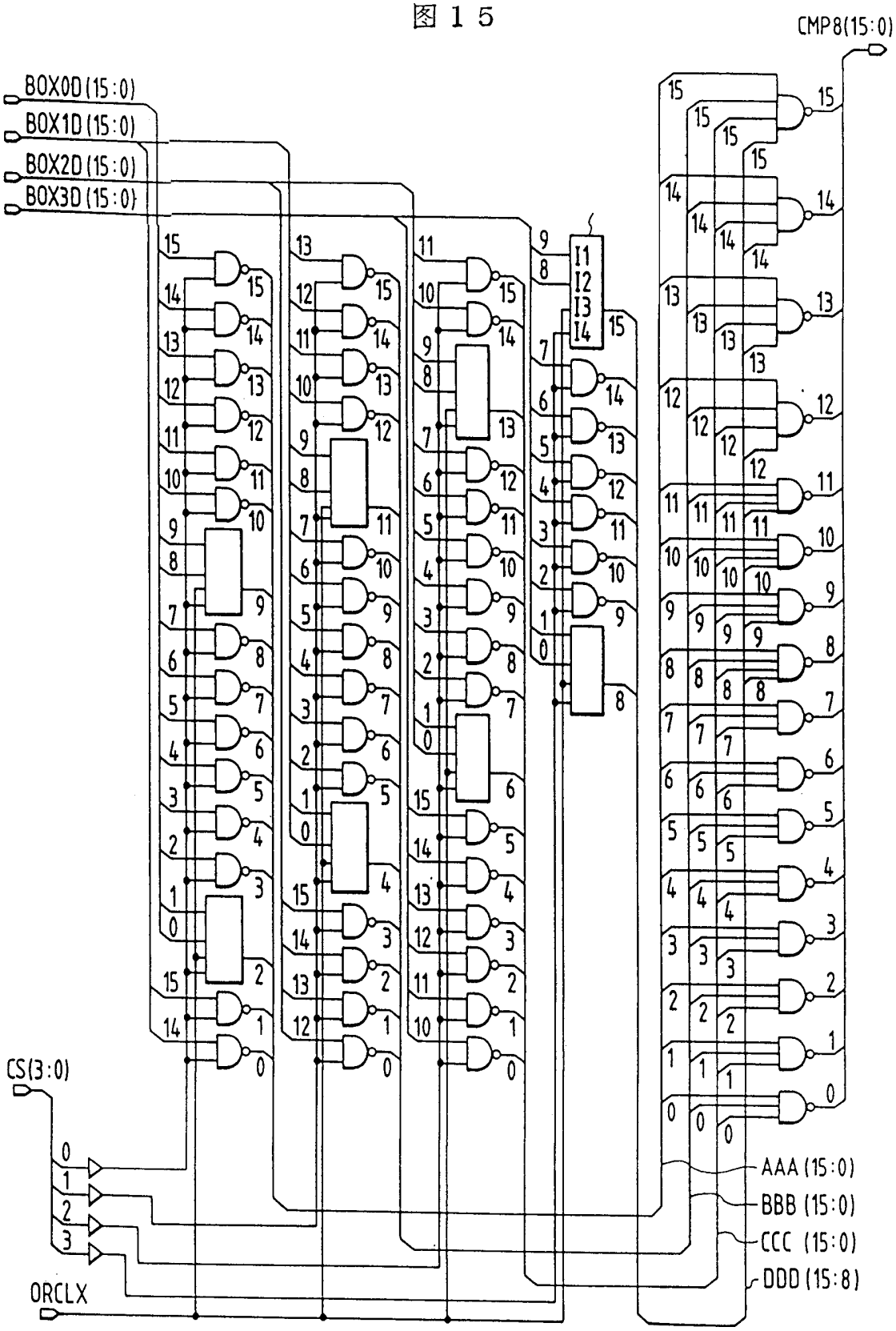

CMP8(15:0)
BOX0D(15:0)
BOX1D(15:0)
BOX2D(15:0)
BOX3D(15:0)
I1
I2
I3
I4
CS(3:0)
ORCLX
AAA(15:0)
BBB(15:0)
CCC(15:0)
DDD(15:8)

图16

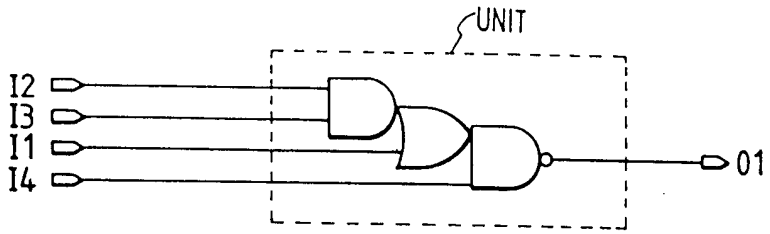


图17

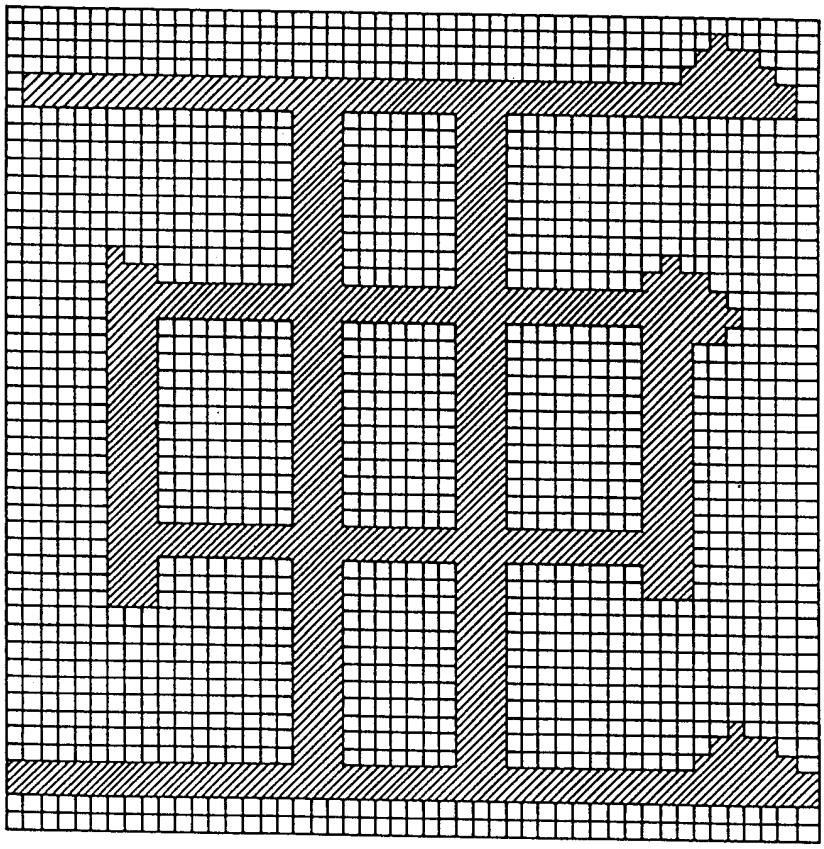


图 1 8

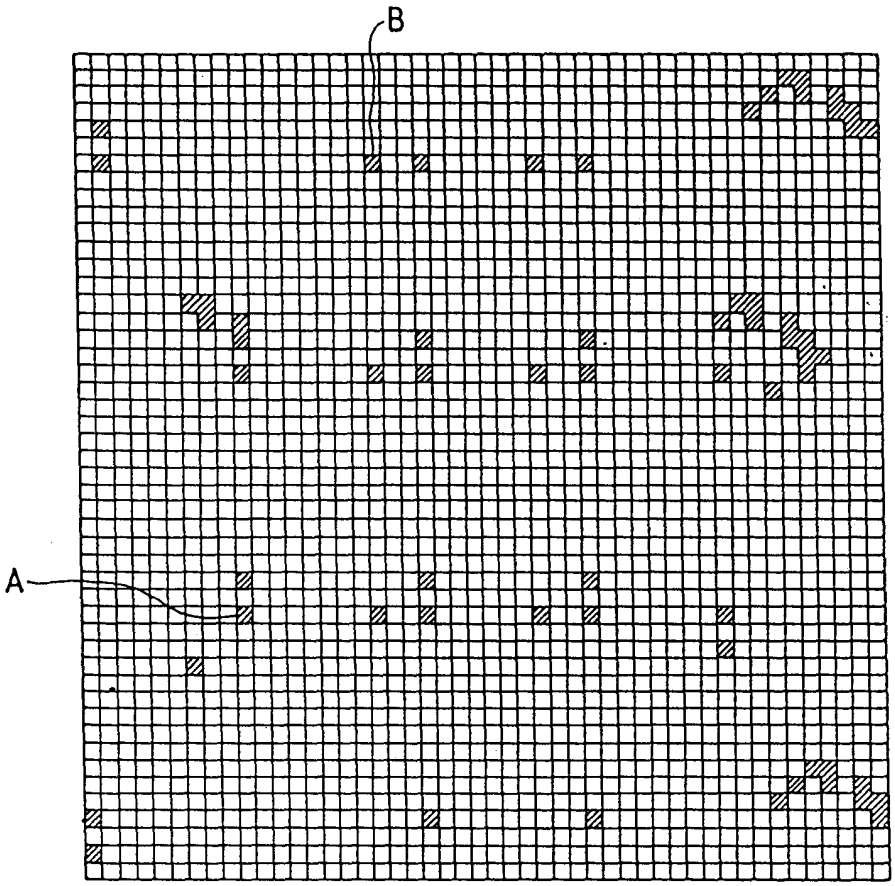
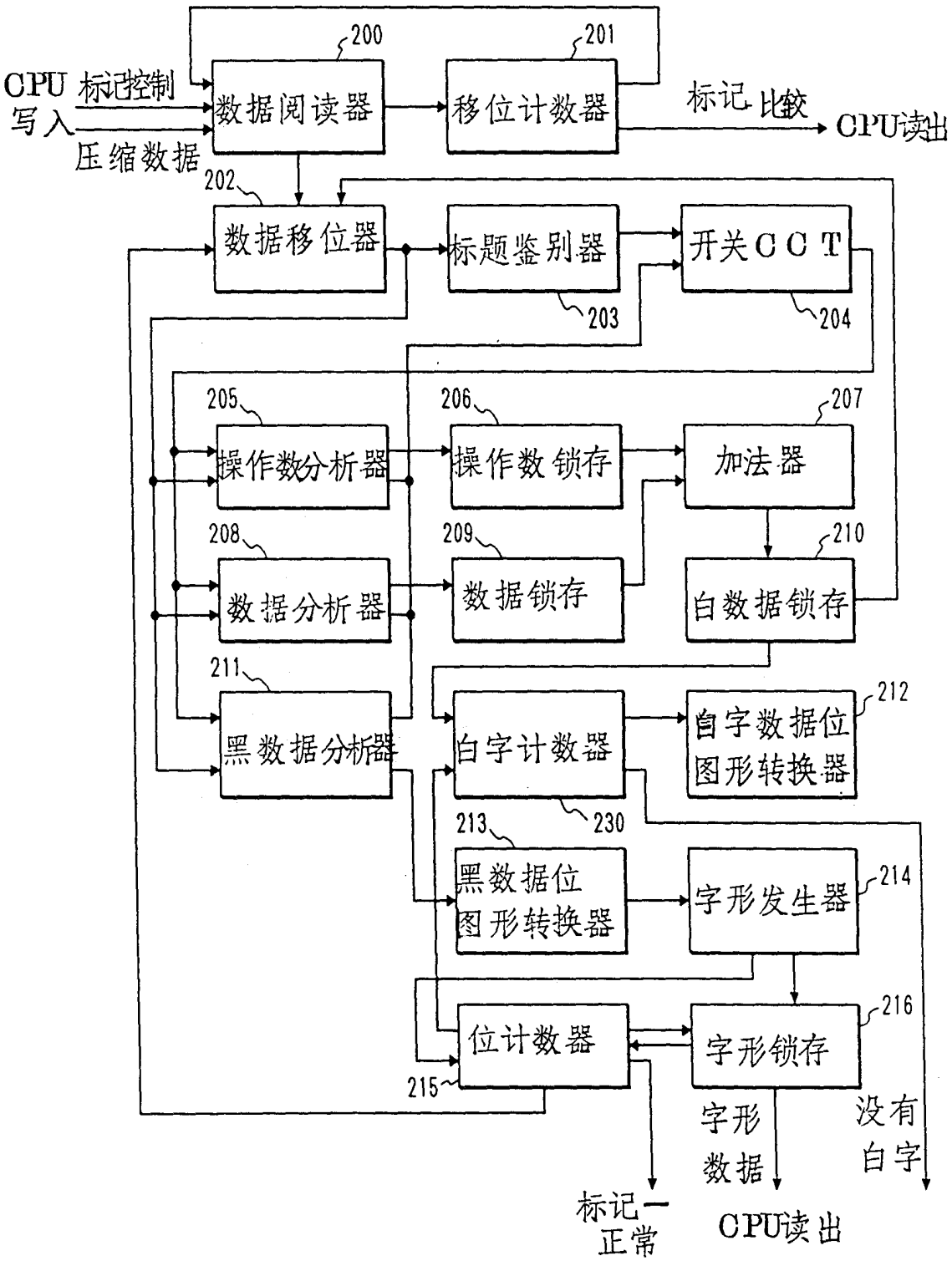


图19

```
$C770
$011C
$72B4
$0532
$51BD
$008D
$A848
$40CA
$8C5A
$546A
$8801
$6554
$2D20
$3546
$00AA
$3818
$48E5
$D38D
$D188
$CDA7
$E252
$9255
$9695
$8B66
$922E
$CB1F
$0002
```

图２０


CPU 标记控制
写入
压缩数据
200
数据阅读器
201
移位计数器
标记比较
CPU读出
202
数据移位器
标题鉴别器
203
开关ＣＣＴ
204
205
操作数分析器
206
操作数锁存
207
加法器
208
数据分析器
209
数据锁存
210
白数据锁存
211
黑数据分析器
白字计数器
212
白字数据位图形转换器
230
213
黑数据位图形转换器
字形发生器
214
位计数器
216
字形锁存
215
字形数据
没有白字
标记正常
CPU读出