(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 107196662 B
(45)授权公告日 2021.07.30

(21)申请号 201710252914.0

(22)申请日 2012.01.12

(65)同一申请的已公布的文献号
申请公布号 CN 107196662 A

(43)申请公布日 2017.09.22

(30)优先权数据
61/432,884 2011.01.14 US

(62)分案原申请数据
201280011782.8 2012.01.12

(73)专利权人 GE视频压缩有限责任公司
地址 美国纽约

(72)发明人 德特勒夫·马佩 通·恩固因 海科·施瓦茨 托马斯·维甘德

(74)专利代理机构 北京康信知识产权代理有限责任公司 11240
代理人 梁丽超 田喜庆

(51)Int.Cl.
H03M 7/40(2006.01)

(56)对比文件
CN 103404035 B,2017.05.10
CN 1112799 A,1995.11.29
CN 1118196 A,1996.03.06
WO 2008129021 A2,2008.10.30
CN 101243459 A,2008.08.13
JP 2010527523 A,2010.08.12
WO 2010052833 A1,2010.05.14
JP 2003153265 A,2003.05.23
JP 2007116501 A,2007.05.10
CN 1158058 A,1997.08.27
CN 1378750 A,2002.11.06
CN 101223573 A,2008.07.16
US 2008165036 A1,2008.07.10
CN 101099391 A,2008.01.02
CN 101198055 A,2008.06.11
CN 1447603 A,2003.10.08
US 5436626 A,1995.07.25
JP 3346204 B2,2002.11.18

审查员 李晓阳

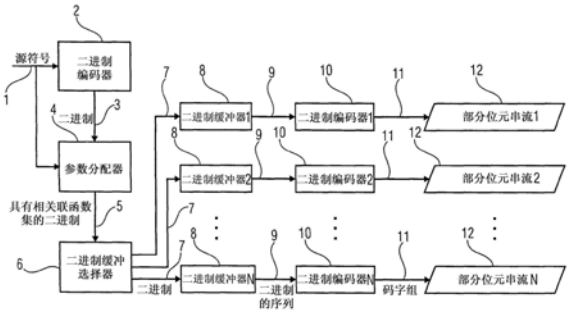
权利要求书8页 说明书62页 附图24页

(54)发明名称

熵编码设备和方法、熵解码设备和方法、及存储介质

(57)摘要

本发明涉及熵编码设备和方法、熵解码设备和方法、及存储介质。将相应语法元素的值范围分解成n个分区的序列，将位于相应分区内的z的分量个别编码，至少一个分量经由VLC编码，且至少一个分量经由PIPE或熵编码，该分解可以适度的编码额外负担极大地增加压缩效率。根据实施例，将语法元素分解成相应数目n的源符号s_i，其中i＝1…n，源符号的相应数目n取决于关于将相应语法元素的值范围再划分成n个分区(140_{1-3})的序列中的哪个分区、相应语法元素的值z属于n个分区的序列中的哪个分区，使得相应数目的源符号s_i的值的和产生z，且若n>1，则对于所有i＝1…n-1而言，s_i的值对应于第i个分区的范围。



CN 107196662 B

1.一种熵编码设备,包含:

一分解器(136),所述分解器被配置为将语法元素的一序列(138)转换成源符号(106)的一序列(106),其中,所述语法元素的序列(138)具有被再划分成N个分区的一序列的值范围,并且其中,所述分解器被配置为通过将所述语法元素的至少一个子组中的每个分别地分解成一相应数目n的源符号s_i来执行所述转换,其中i=1…n,源符号的所述相应数目n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区,以便$s_1+\cdots+s_n$产生z,且若n>1,则对于所有i=1…n-1而言,s_i对应于第i个分区的一范围;

一再划分器(100),所述再划分器被配置为将源符号的所述序列(106)再划分成源符号的一第一子序列(108)及源符号的一第二子序列(110),以使得在所述第一子序列(108)内含有所有源符号sx,其中x为{1…N}的一第一子集的成员,且在所述第二子序列(110)内含有所有源符号sy,其中y为与所述第一子集互斥的{1…N}的一第二子集的成员;

一可变长度码VLC编码器(102),所述VLC编码器被配置为用符号的方式编码所述第一子序列(108)的所述源符号;以及

一算术编码器(104),所述算术编码器被配置为编码源符号的所述第二子序列(110),其中,所述算术编码器使用基于上下文的自适应二进制算术编码。

2.根据权利要求1所述的熵编码设备,其中,所述语法元素的所述子组表示绝对值。

3.根据权利要求2所述的熵编码设备,其中,所述第二子集为{1}。

4.根据权利要求3所述的熵编码设备,其中,N=3。

5.根据权利要求4所述的熵编码设备,其中,所述第一子集为{2,3},其中所述VLC编码器(102)被配置为使用一Golomb-Rice码来用符号的方式编码源符号s2,且使用一指数哥伦布码来用符号的方式编码源符号s3。

6.根据权利要求2所述的熵编码设备,其中,2为所述第一子集的元素,其中所述VLC编码器(102)被配置为使用一Golomb-Rice码来用符号的方式编码源符号s2,且配置为根据先前经编码符号调适所述Golomb-Rice码的一Golomb-Rice参数。

7.根据权利要求1所述的熵编码设备,其中,所述分解器被配置为根据先前经编码源符号调适所述分区之间的一个或更多个极限。

8.根据权利要求1所述的熵编码设备,其中,所述语法元素的所述子组涵盖一图像的变换区块的绝对变换系数的多个绝对变换系数位准,其中一相应变换区块的所述绝对变换系数位准根据引导穿越所述相应变换区块的所述绝对变换系数的一扫描路径排列于语法元素的所述序列(138)内,其中所述分解器被配置为取决于扫描次序居先的所述相应变换区块的绝对变换系数的已编码绝对变换系数位准,或取决于当前欲按所述扫描次序分解的所述绝对变换系数位准的一位置,或基于与当前欲分解的所述绝对变换系数位准的所述位置在空间上或者按扫描次序相邻的变换系数的已编码绝对变换系数位准的一评估,而在分解一相应变换区块的所述绝对变换系数的所述绝对变换系数位准期间调适所述分区之间的一个或更多个极限。

9.根据权利要求1所述的熵编码设备,其中,所述语法元素为不同类型,且所述分解器(136)被配置为取决于所述语法元素的所述类型而执行个别的分解。

10.根据权利要求1所述的熵编码设备,其中,所述算术编码器(104)为一机率区间分割熵编码器,所述机率区间分割熵编码器被配置为编码以字母符号的一序列(124)的形式表

示的源符号的所述第二子序列(110),所述机率区间分割熵编码器包含:

一分配器,所述分配器被配置为基于字母符号的所述序列的先前字母符号内含有的信息,将所述相应字母符号可假定的可能值之中的一机率分布的一估计的一测度分配予字母符号的所述序列中的每一字母符号;

多个熵编码器(116),所述多个熵编码器中的每一熵编码器被配置为将转发至所述相应熵编码器的所述字母符号转换成一相应部分第二串流(118);以及

一选择器(120),所述选择器被配置为将每一字母符号转发至所述多个熵编码器中的选定的一熵编码器(116),所述选择取决于分配予所述相应字母符号的所述测度。

11.根据权利要求10所述的熵编码设备,其中,所述语法元素为不同类型,且所述再划分器(100)被配置为取决于所述语法元素的所述类型而执行所述再划分。

12.根据权利要求10所述的熵编码设备,进一步包含一符号化器(122),所述符号化器被配置为将所述第二子序列(110)的每一源符号个别地映像至字母符号的一相应部分序列中,从而一起形成字母符号的所述序列(124)。

13.根据权利要求12所述的熵编码设备,其中,所述符号化器(122)及所述机率区间分割熵编码器(104)被配置为使得所述字母符号为二进制符号。

14.根据权利要求10所述的熵编码设备,其中,字母符号的所述序列(124)出自一二进制字母,且所述分配器被配置为使得所述机率分布的所述估计由一更小机率或更大机率二进制值的一机率的一估计的一测度、及指定对于两个可能二进制值中的哪个表示所述更小机率或更大机率二进制值的一估计的一识别符组成。

15.根据权利要求10所述的熵编码设备,其中,所述分配器(114)被配置为基于字母符号的所述序列的先前字母符号内含有的信息,将一上下文分配予字母符号的所述序列中的每一字母符号,其中每一上下文具有与所述上下文相关联的一相应机率分布估计,且所述分配器被配置为基于受分配相应上下文的先前字母符号的符号值,使每一上下文的所述机率分布估计适应一实际符号统计学,且所述分配器被配置为基于与分配予所述相应字母符号的所述上下文相关联的所述机率分布估计,决定每一字母符号的所述机率分布的所述估计的所述测度。

16.根据权利要求15所述的熵编码设备,其中,所述分配器(114)被配置为在决定每一字母符号的所述机率分布的所述估计的所述测度时,对多个机率分布估计代表中的一个机率分布估计代表量化与分配予所述相应字母符号的所述上下文相关联的所述机率分布估计,以获取所述机率分布的所述估计的所述测度,且其中所述选择器(120)被配置为使得自所述多个机率分布估计代表至所述多个熵编码器(116)定义出一满射关联。

17.根据权利要求16所述的熵编码设备,其中,所述选择器(114)被配置为取决于字母符号的所述序列(124)的先前字母符号,以一预定决定性方式随时间改变自所述机率分布估计的一范围至所述多个机率分布估计代表的一量化映射。

18.根据权利要求17所述的熵编码设备,其中,所述多个熵编码器(116)被配置为响应量化映射的一改变,调适所述多个熵编码器(116)将字母符号转换成所述部分第二串流的方式。

19.根据权利要求17所述的熵编码设备,其中,所述选择器(114)被配置为改变量化映射,以使得所述熵编码器(116)将所述字母符号转换成的所述部分第二串流(118)的位率显

得不太分散。

20.根据权利要求10所述的熵编码设备,其中,所述多个熵编码器(116)中的至少一个熵编码器具有与所述至少一个熵编码器相关联的一符号输入缓冲器,其中所述选择器(120)被配置为经由所述相关联符号输入缓冲器将所述字母符号转发至所述至少一个熵编码器(116)。

21.根据权利要求10所述的熵编码设备,其中,所述多个熵编码器(116)为可变长度编码器,所述可变长度编码器被配置为将字母符号序列映像至码字组。

22.根据权利要求10所述的熵编码设备,其中,所述多个熵编码器(116)中的每一熵编码器为一可变长度编码器,所述可变长度编码器被配置为将可变长度的字母符号序列映像至固定长度的码字组。

23.一种熵解码设备,包含:

一VLC解码器(200),所述VLC解码器被配置为用码字组的方式自一第一位串流的码字组重建源符号的一第一子序列(204)的源符号;

一算术解码器(202),所述算术解码器(202)被配置为重建源符号的一第二子序列(208),其中,所述算术解码器使用基于上下文的自适应二进制算法解码;

一组合器(224),所述组合器被配置为自源符号的所述第一子序列(204)及源符号的所述第二子序列(208)组合语法元素的一序列(226),其中,所述语法元素的序列(226)具有被再划分成N个分区的一序列的值范围,并且其中,所述组合器被配置为:通过针对所述语法元素的至少一个子组中的每个,决定源符号s_i的相应数目n来从所述相应数目n的源符号分别地组合每个语法元素,以便执行所述组合,其中i=1…n取决于所述相应语法元素的一值z属于所述相应语法元素的一值范围再划分成的N个分区的一序列中的哪个分区,只要s_i对应于第i个分区的一范围则通过根据$s_1+\cdots+s_n$自1至N加总所述相应数目的源符号s_i来获取所述语法元素的值z,其中所述组合器(224)被配置为自所述第一子序列(204)检索所有源符号sx,其中x为{1…N}的一第一子集的成员,且所述组合器被配置为自所述第二子序列(208)检索所有源符号sy,其中y为与所述第一子集互斥的{1…N}的一第二子集的成员。

24.根据权利要求23所述的熵解码设备,其中,所述语法元素的所述子组表示绝对值。

25.根据权利要求24所述的熵解码设备,其中,所述第二子集为{1}。

26.根据权利要求25所述的熵解码设备,其中,N=3。

27.根据权利要求26所述的熵解码设备,其中,所述第一子集为{2,3},其中所述VLC解码器(200)被配置为使用一Golomb-Rice码来用码字组的方式重建源符号s2,且使用一指数哥伦布码来用码字组的方式重建源符号s3。

28.根据权利要求23所述的熵解码设备,其中,2为所述第一子集的元素,其中所述VLC解码器(102)被配置为使用一Golomb-Rice码来用码字组的方式重建源符号s2,且被配置为根据先前经重建源符号调适所述Golomb-Rice码的一Golomb-Rice参数。

29.根据权利要求23所述的熵解码设备,所述熵解码设备进一步包含一复合器(220),所述复合器被配置为再结合源符号的所述第一子序列(204)及源符号的所述第二子序列,以获取源符号的所述序列(218)。

30.根据权利要求23所述的熵解码设备,其中,所述语法元素为不同类型,且所述组合器被配置为取决于所述语法元素的所述类型而执行个别的组合。

31.根据权利要求23所述的熵解码设备，其中，所述语法元素的所述子组涵盖一图像的变换区块的绝对变换系数的绝对变换系数位准，其中一相应变换区块的所述绝对变换系数位准根据引导穿越所述相应变换区块的所述绝对变换系数的一扫描路径排列于语法元素的所述序列(138)内，其中所述组合器被配置为取决于扫描次序居先的所述相应变换区块的绝对变换系数的已重建绝对变换系数位准，或取决于当前欲按所述扫描次序组合的所述绝对变换系数位准的一位置，或基于与当前欲组合的所述绝对变换系数位准的所述位置在空间上或者按扫描次序相邻的变换系数的所述已重建绝对变换系数位准的一评估，而在组合一相应变换区块的所述绝对变换系数的所述绝对变换系数位准期间调适所述分区之间的一个或更多个极限。

32.根据权利要求24所述的熵解码设备，所述组合器被配置为根据先前重建源符号调适所述分区之间的一个或更多个极限。

33.根据权利要求23所述的熵解码设备，其中，所述算术解码器是被配置为重建源符号的第二序列(110)的机率区间分割熵解码器，所述源符号的第二序列(110)以字母符号的一序列(224)的形式表示，且响应顺序地请求所述字母符号的字母符号请求，所述机率区间分割熵解码器包含：

多个熵解码器(210)，所述多个熵解码器中的每一熵解码器被配置为将一相应部分第二串流(216)转换成字母符号的所述序列的字母符号；

一分配器(212)，所述分配器被配置为基于字母符号的所述序列的先前重建的字母符号内含有的信息，将所述相应字母符号可假定的可能值之中一机率分布的一估计的一测度，分配予对于表示待重建的源符号的所述第二子序列(208)的字母符号的所述序列中的一字母符号的每一请求；以及

一选择器(214)，所述选择器被配置为针对表示待重建的源符号的所述第二子序列(208)的字母符号的所述序列的一字母符号的每一请求，自所述多个熵解码器(210)中选定的一熵解码器检索字母符号的所述序列中的所述相应字母符号，而其选择取决于分配予针对所述相应字母符号的所述相应请求的所述测度，

其中源符号的所述第一子序列(204)及源符号的所述第二子序列(208)共通地形成源符号的一序列(218)。

34.根据权利要求33所述的熵解码设备，其中，源符号的所述序列(218)为一可剖析位串流的语法元素的一序列。

35.根据权利要求34所述的熵解码设备，其中，所述语法元素为不同类型，且所述第一位串流(206)及所述部分第二串流(216)彼此交错成一经交错位串流(228)，所述熵解码设备被配置为取决于所述语法元素的所述类型而分隔所述第一位串流(206)与所述部分第二串流(216)。

36.根据权利要求33所述的熵解码设备，所述熵解码设备进一步包含一去符号化器(222)，所述去符号化器被配置为以字母符号的部分序列的多个单元，将字母符号的所述序列(224)个别地再映射至所述第二子序列(208)的所述源符号中。

37.根据权利要求36所述的熵解码设备，其中，所述去符号化器(222)及所述机率区间分割熵解码器(202)被配置为使得所述字母符号为二进制符号。

38.根据权利要求33所述的熵解码设备，其中字母符号的所述子序列出自一二进制字

母，且所述分配器被配置为使得所述机率分布的所述估计由所述二进制字母的两个可能二进制值中的一更小机率或更大机率二进制值的一机率的一估计的一测度、及指定对于所述两个可能二进制值中的哪个表示所述更小机率或更大机率二进制值的一估计的一识别符组成。

39.根据权利要求33所述的熵解码设备，其中，所述分配器(212)被配置为基于字母符号的所述子序列的先前重建的字母符号内含有的信息，在内部将一上下文分配予字母符号的所述子序列中的一字母符号的每一请求，其中每一上下文具有与所述上下文相关联的一相应机率分布估计，且所述分配器被配置为基于受分配所述相应上下文的先前重建的字母符号的符号值，使每一上下文的所述机率分布估计适应一实际符号统计学，且所述分配器被配置为基于与分配予所述相应字母符号的所述上下文相关联的所述机率分布估计，针对字母符号的所述子序列的一字母符号的每一请求，决定字母符号的所述子序列的所述相应字母符号的所述机率分布的所述估计的所述测度。

40.根据权利要求39所述的熵解码设备，其中，所述分配器(212)被配置为在决定一字母符号的每一请求的所述机率分布的所述估计的所述测度时，对多个机率分布估计代表中的一个机率分布估计代表量化与随所述相应字母符号分配的所述上下文相关联的所述机率分布估计，以获取所述机率分布的所述估计的所述测度，且其中所述选择器被配置为使得自所述多个机率分布估计代表至所述多个熵解码器定义出一满射关联。

41.根据权利要求40所述的熵解码设备，其中，所述选择器被配置为取决于字母符号的所述序列的先前重建的字母符号，以一预定决定性方式随时间改变自所述机率分布估计的一范围至所述多个机率分布估计代表的一量化映射。

42.根据权利要求41所述的熵解码设备，其中，所述多个熵解码器被配置为响应量化映射的一改变，调适所述多个熵解码器将所述相应部分第二串流转换成字母符号的方式。

43.根据权利要求41所述的熵解码设备，其中，所述选择器被配置为改变量化映射，以使得自所述多个熵解码器检索所述字母符号的比率变得不太分散。

44.根据权利要求33所述的熵解码设备，其中，至少一个熵解码器具有与所述至少一个熵解码器相关联的一符号输出缓冲器，其中所述选择器被配置为经由所述相关联符号输出缓冲器自所述至少一个熵解码器检索所述字母符号。

45.根据权利要求33所述的熵解码设备，其中，所述熵解码器为可变长度解码器，所述可变长度解码器被配置为将码字组映像至字母符号序列。

46.根据权利要求45所述的熵解码设备，所述熵解码设备进一步包含一去交错器(230)，所述去交错器被配置为经由按照由源符号的所述序列(218)内的所述第一子序列(204)及所述第二子序列(208)的所述源符号的次序所定义的一次序，使一单一位串流(228)内的码字组去交错，来产生所述第一位串流及所述部分第二串流。

47.根据权利要求46所述的熵解码设备，其中，所述去交错器(230)被配置为，

每当目前成直线的源符号的所述序列(218)内的一源符号属于所述第一子序列(204)时，将单一位串流(228)内的一当前码字组视为所述第一位串流(206)的一码字组，且

每当目前成直线的源符号的所述序列(218)内的一源符号属于所述第二子序列(208)时，将单一位串流(228)内的一当前码字组视为所述部分第二串流(216)中的一相应部分第二串流的一码字组，若属于所述当前源符号的所述字母符号中的任一字母符号使所述相应

部分第二串流(216)的一码字组至一相应字母符号序列的一新的映射成为必需,所述新的映射经由取决于分配予所述相应字母符号的所述请求的所述测度而选定的所述熵解码器(210),则在属于所述当前源符号的所述字母符号中的若干字母符号存在的情况下使用所述字母符号之中的一次序,且所述次序是取决于字母符号的所述序列(224)内的所述字母符号的所述次序。

48.一种熵编码方法,包含以下步骤:

将语法元素的一序列(138)转换成源符号(106)的一序列(106),其中,所述语法元素的序列(138)具有被再划分成N个分区的一序列的值范围,并且其中,所述转换包括将所述语法元素的至少一个子组中的每个分别地分解成一相应数目N的源符号s_i,其中i=1…n,源符号的所述相应数目n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区,以便$s_1+\cdots+s_n$产生z,且若n>1,则对于所有i=1…n-1而言,s_i对应于第i个分区的一范围;

将源符号的所述序列(106)再划分成源符号的一第一子序列(108)及源符号的一第二子序列(110),以使得在所述第一子序列(108)内含有所有源符号sx,其中x为{1…N}的一第一子集的成员,且在所述第二子序列(110)内含有所有源符号sy,其中y为与所述第一子集互斥的{1…N}的一第二子集的成员;

经由VLC编码,用符号的方式编码所述第一子序列(108)的所述源符号;以及

经由使用基于上下文的自适应二进制算术编码的算术编码,编码源符号的所述第二子序列(110)。

49.一种熵解码方法,包含以下步骤:

经由VLC解码,用码字组的方式自一第一位串流(206)的码字组重建源符号的一第一子序列(204)的源符号;

经由使用基于上下文的自适应二进制算术解码的算术解码,重建源符号的一第二子序列(208);

自源符号的所述第一子序列(204)及源符号的所述第二子序列(208)组合语法元素的一序列(226),其中,所述语法元素的序列(226)具有被再划分成N个分区的一序列的值范围,并且其中,所述组合包括通过针对所述语法元素的至少一个子组中的每个决定源符号s_i的相应数目N,来从所述相应数目n的源符号分别地组合每个语法元素,其中i=1…n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区,只要s_i对应于第i个分区的一范围则通过根据$s_1+\cdots+s_n$自1至N加总所述相应数目的源符号s_i来获取所述语法元素的值z;其中所述组合(224)包括自所述第一子序列(204)检索所有源符号sx,其中x为{1…N}的一第一子集的成员,且所述组合(224)包括自所述第二子序列(208)检索所有源符号sy,其中y为与所述第一子集互斥的{1…N}的一第二子集的成员。

50.一种数字存储介质,具有存储在其上的由根据权利要求48所述的熵编码方法编码的压缩信号。

51.一种数字存储介质,具有存储在其上的压缩信号,所述压缩信号包括源符号的一第一子序列(108)和源符号的一第二子序列(110),

其中,源符号的所述第一子序列(108)与源符号的所述第二子序列(110)一起构成源符号(106)的序列(106),其中,语法元素的一序列(138)被转换成源符号(106)的一序列

(106)，其中，所述语法元素的序列(138)具有被再划分成N个分区的一序列的值范围，通过将所述语法元素的至少一个子组中的每个分别地分解成一相应数目n的源符号s_i来执行所述转换，其中i＝1…n，源符号的所述相应数目n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区，以便s_1+…+s_n产生z，且若n>1，则对于所有i＝1…n-1而言，s_i对应于第i个分区的一范围；

其中，源符号的所述序列(106)被再划分成源符号的所述第一子序列(108)及源符号的所述第二子序列(110)，以使得在所述第一子序列(108)内含有所有源符号sx，其中x为{1…N}的一第一子集的成员，且在所述第二子序列(110)内含有所有源符号sy，其中y为与所述第一子集互斥的{1…N}的一第二子集的成员；

其中，通过符号明智编码对所述第一子序列(108)的所述源符号进行VLC编码；以及

其中，源符号的所述第二子序列(110)使用基于上下文的自适应二进制算术编码进行算术编码。

52.根据权利要求50或51所述的数字存储介质，其中，所述压缩信号以可扩展的方式编码视频。

53.用于解码压缩信号的方法，其中，所述方法包括：

接收并解码通过根据权利要求48所述的熵编码方法编码的压缩信号。

54.用于解码压缩信号的方法，其中，所述方法包括：

接收并解码压缩信号，所述压缩信号包括源符号的一第一子序列(108)和源符号的一第二子序列(110)，

其中，源符号的所述第一子序列(108)与源符号的所述第二子序列(110)一起构成源符号(106)的序列(106)，其中，语法元素的一序列(138)被转换成源符号(106)的一序列(106)，其中，所述语法元素的序列(138)具有被再划分成N个分区的一序列的值范围，通过将所述语法元素的至少一个子组中的每个分别地分解成一相应数目n的源符号s_i来执行所述转换，其中i＝1…n，源符号的所述相应数目n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区，以便s_1+…+s_n产生z，且若n>1，则对于所有i＝1…n-1而言，s_i对应于第i个分区的一范围；

其中，源符号的所述序列(106)被再划分成源符号的所述第一子序列(108)及源符号的所述第二子序列(110)，以使得在所述第一子序列(108)内含有所有源符号sx，其中x为{1…N}的一第一子集的成员，且在所述第二子序列(110)内含有所有源符号sy，其中y为与所述第一子集互斥的{1…N}的一第二子集的成员；

其中，通过符号明智编码对所述第一子序列(108)的所述源符号进行VLC编码；以及

其中，源符号的所述第二子序列(110)使用基于上下文的自适应二进制算术编码进行算术编码。

55.用于存储压缩信号的方法，包括：

在数字存储介质上存储压缩信号，所述压缩信号包括源符号的一第一子序列(108)和源符号的一第二子序列(110)，

其中，源符号的所述第一子序列(108)与源符号的所述第二子序列(110)一起构成源符号(106)的序列(106)，其中，语法元素的一序列(138)被转换成源符号(106)的一序列(106)，其中，所述语法元素的序列(138)具有被再划分成N个分区的一序列的值范围，通过

将所述语法元素的至少一个子组中的每个分别地分解成一相应数目n的源符号s_i来执行所述转换，其中i＝1…n，源符号的所述相应数目n取决于所述相应语法元素的一值z属于N个分区的一序列中的哪个分区，以便s_1+…+s_n产生z，且若n>1，则对于所有i＝1…n-1而言，s_i对应于第i个分区的一范围；

其中，源符号的所述序列(106)被再划分成源符号的所述第一子序列(108)及源符号的所述第二子序列(110)，以使得在所述第一子序列(108)内含有所有源符号sx，其中x为{1…N}的一第一子集的成员，且在所述第二子序列(110)内含有所有源符号sy，其中y为与所述第一子集互斥的{1…N}的一第二子集的成员；

其中，通过符号明智编码对所述第一子序列(108)的所述源符号进行VLC编码；以及

其中，源符号的所述第二子序列(110)使用基于上下文的自适应二进制算术编码进行算术编码。

56.根据权利要求53至55中任一项所述的方法，其中，所述压缩信号以可扩展的方式编码视频。

熵编码设备和方法、熵解码设备和方法、及存储介质

[0001] 本申请为国际申请日为2012年1月12日、国际申请号为PCT/EP2012/050431、发明名称为“熵编码及解码方案”的中国国家阶段申请的分案申请，该中国国家阶段申请的进入国家阶段日为2013年9月4日、申请号为201280011782.8、发明名称为“熵编码及解码方案”。

技术领域

[0002] 本发明是关于熵编码及解码，且本发明可用于诸如视讯及音讯压缩的应用中。

背景技术

[0003] 熵编码一般可视为无损数据压缩的最一般的形式。无损压缩目的在于用比原始数据表示所需要的位更少的位显示离散数据，但无任何信息损失。离散数据可以用本文、图形、影像、视讯、音讯、语音、传真、医疗数据、气象数据、财务数据或任何其它形式的数字数据的形式给出。

[0004] 在熵编码中，通常忽略下层离散数据源的特定高阶特性。因此，将任何数据源视为作为源符号的序列给出，源符号的该序列用给定的m阶字母取值，且源符号的该序列的特征在于相应的(离散)机率分布$\{p_1,\cdots,p_m\}$。在该抽象设定中，依据每符号位的预期码字组长度任何熵编码方法的下限经由以下熵来给定：

[0005] $$H=-\sum_{i=1}^{m}p_i\log_2 p_i\,.\quad(A1)$$

[0006] 霍夫曼(Huffman)码及算术码为能够接近熵极限(在某种意义上)的实用码的熟知实例。对于固定机率分布而言，霍夫曼码相对容易建构。霍夫曼码的最吸引人的性质为，该霍夫曼码的实施可有效率地经由使用可变长度码(VLC)表来实现。然而，当处理时变源统计学(亦即，变化的符号机率)时，极其需要调适霍夫曼码及该霍夫曼码的相应的VLC表，霍夫曼码及该霍夫曼码的相应的VLC表两者皆依据算法复杂性以及依据实施成本。又，在具有主要字母值，其中$p_k>0.5$的情况下，相应的霍夫曼码的冗余度(不使用诸如游程编码的任何字母延伸)可能为相当实质的。霍夫曼码的另一缺陷经由在处理高阶机率模型化情况下可能需要多组VLC表的事实给出。另一方面，尽管算术编码实质上比VLC更复杂，但当应对适应性且高阶机率模型化以及应对高度偏斜机率分布的情况时，该算术编码提供更一致且更充分处理的优点。实际上，此特性基本上由算术编码提供将机率估计的任何给定值以差不多直接的方式映像至所得码字组的部分的至少概念性机制的事实产生。在具备此接口的情况下，算术编码允许一方面机率模型化及机率估计的任务与另一方面实际的熵编码亦即将符号映像至码字组之间的清晰分离。

[0007] 算术编码及VLC编码的替代物为PIPE编码。更精确而言，在PIPE编码中，单位区间分割成互斥机率区间的小集合，以沿着随机符号变量的机率估计处理的编码设计PIPE。根据此分割，可将具有任意字母大小的离散源符号的输入序列映像至字母符号的序列，且将该字母符号中的每一字母符号分配给一个特定机率区间，该特定机率区间又经由尤其专用的熵编码程序来编码。在该区间中的每一区间由固定机率表示的情况下，机率区间分割熵

(PIPE)编码程序可基于简单可变对可变长度码的设计及应用。机率模型化可为固定的或者适应性的。然而,尽管PIPE编码显著地不如算术编码复杂,但该PIPE编码仍具有比VLC编码更高的复杂性。

[0008] 因此,将有利的是,具有所探讨的能够实现一方面编码复杂性与另一方面压缩效率之间的最佳取舍的熵编码方案,甚至当与PIPE编码相比较时,该熵编码方案已组合算术编码与VLC编码两者的优点。

[0009] 此外,一般而言,将有利的是,具有所探讨的能够以适度的编码复杂性实现自身最佳压缩效率的熵编码方案。

发明内容

[0010] 因此,本发明的目标为,提供实现以上经识别的需求亦即能够实现一方面编码复杂性与另一方面压缩效率之间的最佳取舍的熵编码概念。

[0011] 此目标经由根据权利要求1项或第23项的设备、根据权利要求49项或第50项的方法及根据权利要求51项的计算机程序来实现。

[0012] 本发明基于以下想法:将相应语法元素的值范围分解成n个分区的序列,其中将单独放置于相应分区内的语法元素值z的分量编码,其中至少一个分量经由VLC编码,且至少一个分量经由PIPE或算术编码或任何其它熵编码方法,该分解可以适度的编码额外负担极大地增加压缩效率,因为所使用的编码方案可更好地适应于语法元素统计学。相应地,根据本发明的实施例,将语法元素分解成相应数目n的源符号s_i,其中i=1…n,源符号的相应数目n取决于将相应语法元素的值范围再划分成关于n个分区(140_{1-3})的序列的哪个分区、相应语法元素的值z属于n个分区(140_{1-3})的序列的哪个分区,以便相应数目的源符号s_i的值的和产生z,且若n>1,则对于所有i=1…n-1而言,s_i的值对应于第i个分区的范围。

[0013] 此外,本发明的一些实施例使用除VLC编码之外的PIPE编码,该实施例利用以下事实:若除PIPE编码的分类特征之外,提供进一步分类阶段,则可实现一方面编码复杂性与另一方面压缩效率之间的较佳取舍,根据PIPE编码的该分类特征,字母符号分布于根据该字母符号的机率分布估计的多个专门化熵编码/解码器之中,根据该进一步分类阶段,待编码的源符号再划分成经受VLC编码的第一子串流及经受PIPE编码亦即用PIPE编码器的符号字母表示为字母符号的序列的第二子串流。经由此测度,具有适当的符号机率分布(亦即,适合于借助于VLC编码来有效率地编码而不具有以上本申请案的说明书的导论部分中概括的缺陷的机率分布)的源符号可分类为诸如较高取值分区的VLC编码符号,而诸如最低取值分区的其它源符号可视为PIPE编码符号且经受PIPE编码,该PIPE编码的编码复杂性比VLC编码更高,但该PIPE编码具有更佳的压缩效率。

[0014] 根据实施例,有可能将语法元素分解成相应整数数目的源符号,该源符号中的一个源符号视为VLC编码符号,该源符号中的另一个源符号视为PIPE编码符号。

[0015] 本发明的较佳态样为封闭的附属项的主题。

附图说明

[0016] 参阅以下示图描述本发明的较佳实施例,在该示图之中

[0017] 图1A图示根据实施例的熵编码设备的方块图;

[0018] 图1B图示说明根据实施例,可能将语法元素分解成源符号的示意图;

[0019] 图1C图示说明根据实施例,图1A的分解器在将语法元素分解成源符号时可能的操作模式的流程图;

[0020] 图2A图标根据实施例的熵解码设备的方块图;

[0021] 图2B图示说明根据实施例,图2A的组合器在自源符号组合语法元素时的可能操作模式的流程图;

[0022] 图3图示根据实施例的可用于图1A-图1C的PIPE编码器的方块图;

[0023] 图4图示根据实施例的可用于图2A和图2B的适合于解码由图3的PIPE编码器产生的位串流的PIPE解码器的方块图;

[0024] 图5图示说明根据实施例的具有多任务部分位串流的数据封包的示意图;

[0025] 图6图示说明根据另一实施例的具有使用固定大小区段的替代性分段的资料封包的示意图;

[0026] 图7图标根据实施例的使用部分位串流交错的PIPE编码器的方块图;

[0027] 图8A和图8B图示根据实施例,图7的编码器侧的码字组缓冲器的状态的示意性说明实例;

[0028] 图9图标根据实施例的使用部分位串流交错的PIPE解码器的方块图;

[0029] 图10图示根据实施例的使用码字组的PIPE解码器的方块图,该码字组使用单组码字组来交错;

[0030] 图11图示根据实施例的使用固定长度位序列的交错的PIPE编码器的方块图;

[0031] 图12A和图12B图示根据实施例,图11的编码器侧的全局位缓冲器的状态的示意性说明实例;

[0032] 图13图示根据实施例的使用固定长度位序列的交错的PIPE解码器的方块图;

[0033] 图14图示说明假定在(0,0.5]内均匀机率分布,成为K＝4区间的最佳机率区间离散化的图形;

[0034] 图15图示说明p＝0.38的LPB机率及经由霍夫曼算法获取的相关联可变长度码的二元事件树的示意图;

[0035] 图16图标图形,自该图形可收集给定最大数目的表项L_m的最佳码的相对位率增加ρ(p,C);

[0036] 图17图示说明理论最佳机率区间分割成K＝12区间的速率增加及具有最大数目的L_m＝65表项的V2V码的实际设计的图形;

[0037] 图18图示说明将三元选择树转换成全二元选择树的实例的示意图;

[0038] 图19图标根据实施例的系统的方块图,该系统包含编码器(左部)及解码器(右部);

[0039] 图20图标根据实施例的熵编码设备的方块图;

[0040] 图21图标根据实施例的熵解码设备的方块图;

[0041] 图22图示根据甚至进一步实施例的熵编码设备的方块图;

[0042] 图23A和图23B图示说明根据实施例,图22的编码器侧的全局位缓冲器的状态的实例的示意图;

[0043] 图24图示根据本申请案的另一实施例的熵解码设备的方块图。

具体实施方式

[0044] 在关于以下示图中描述本申请案的若干实施例之前，注意到，在整个示图中使用相同组件符号以表示该示图中相同或等效组件，且只要先前描述不与当前示图的描述冲突，则用先前示图中的任一示图提供的该组件的描述亦应应用于以下示图中的任一示图。

[0045] 图1A图示根据本申请案的实施例的熵编码设备。该设备包含再划分器100、VLC编码器102及PIPE编码器104。

[0046] 再划分器100经配置以将源符号106的序列再划分成源符号的第一子序列108及源符号的第二子序列110。VLC编码器102具有该VLC编码器102的输入，该输入连接至再划分器100的第一输出，且VLC编码器102经配置以将第一子序列108的源符号以符号的方式转换成形成第一位串流112的码字组。VLC编码器102可包含检查表且个别地使用作为标示的源符号，以在检查表中每源符号查找相应码字组。VLC编码器输出后者码字组，且VLC编码器在子序列110中继续进行以下源符号，以输出码字组的序列，在码字组的该序列中每一码字组与子序列110内源符号中的一个源符号精确地相关联。码字组可具有不同长度，且码字组可经定义以使得无码字组形成具有其它码字组中的任一码字组的前缀。此外，该检查表可为静态的。

[0047] PIPE编码器104具有该PIPE编码器104的输入，该输入连接至再划分器100的第二输出，且PIPE编码器104经配置以编码以字母符号的序列的形式表示的源符号的第二子序列110，且PIPE编码器104包含分配器114、多个熵编码器116及选择器120，该分配器114经配置以基于字母符号的序列的先前字母符号内含有的信息，将相应字母符号可假定的可能值之中的机率分布估计的测度分配予字母符号的序列中的每一字母符号，该多个熵编码器116中的每一熵编码器116经配置以将转发至相应熵编码器的字母符号转换成相应第二位串流118，该选择器120经配置以将第二子序列110中的每一字母符号转发至该多个熵编码器116中的选定的熵编码器116，该选择取决于分配予相应字母符号的机率分布的估计的先前提及的测度。源符号与字母符号之间的关联可为使得每一字母符号唯一地与子序列110的恰一个源符号相关联，以与可能紧接于彼此的后的字母符号的序列的可能地另外字母符号一起表示此一个源符号。

[0048] 如以下更详细地描述，源符号的序列106可为可剖析位串流的语法元素的序列。例如，可剖析位串流可用语法元素以可扩充或非可扩充方式表示视讯及/或音讯内容，该语法元素表示(例如)变换系数位准、移动向量、电影参考指标、比例因子、音讯包封能量值或类似物。特定言的，语法元素可为具有相同类型的语法元素的不同类型或类别，例如，在可剖析位串流内但相对于该可剖析位串流的不同部分(诸如，不同图像、不同宏区块、不同光谱分量或类似物)具有相同意义，而不同类型的语法元素可在位串流内具有不同意义，诸如，移动向量具有与表示变换系数位准的语法元素不同的意义，该变换系数位准表示移动预测残留。

[0049] 再划分器100可经配置以取决于语法元素的类型而执行再划分。亦即，再划分器100可将第一组类型的语法元素转发至第一子序列108且将不同于第一组的第二组类型的语法元素转发至第二子序列110。经由再划分器100执行的再划分可设计为使得子序列108内语法元素的符号统计学适合于经由VLC编码器102来VLC编码，亦即，尽管使用VLC编码及该VLC编码关于该VLC编码对于本申请案的说明书的导论部分中概括的某些符号统计学的

适合性的限制，皆产生几乎可能的最小熵。另一方面，再划分器100可将所有其它语法元素转发至第二子序列110上，以便经由更复杂但更有效率(依据压缩比)的PIPE编码器104来编码具有不适合于VLC编码的符号统计学的该语法元素。

[0050] 亦如以下示图的更详细描述的实施例的情况，PIPE编码器104可包含符号化器122，该符号化器122经配置以将第二子序列110的每一语法元素个别地映像至字母符号的相应部分序列中，从而一起形成先前提及的字母符号的序列124。换言的，例如，若已将子序列110的源符号表示为字母符号的相应部分序列，则可能不提供符号化器122。例如，符号化器122在子序列110内源符号为不同字母且尤其具有不同数目的可能字母符号的字母的情况下是有利的。亦即，在此情况下，符号化器122能够协调到达子串流110内的符号的字母。例如，符号化器122可实施为二进制编码器，该二进制编码器经配置以二进制化到达子序列110中的符号。

[0051] 如之前所提及，语法元素可为不同类型。此状况亦可对子串流110内的语法元素成立。符号化器122可随后经配置以使用符号化映像方案(诸如二进制化方案)执行子序列110的语法元素的个别映像，对于不同类型的语法元素而言，方案有所不同。特定二进制化方案的实例存在于以下描述中，诸如，一元的二进制化方案、0阶或1阶(例如，或截断式一元的二进制化方案)指数哥伦布(exp-Golomb)二进制化方案、截断式且重新排序的指数哥伦布0阶二进制化方案或非系统二进制化方案。

[0052] 相应地，熵编码器116可经配置以对二进制字母表操作。最终，应注意，符号化器122可视为图1A中所示的PIPE编码器104本身的部分。然而，或者，二进制编码器可视为在PIPE编码器的外部。

[0053] 类似于后者通知，应注意，尽管分配器114图示为串行连接于符号化器122与选择器120之间，但分配器114可替代性地视为连接于符号化器124的输出与选择器120的第一输入之间，其中分配器114的输出连接至随后关于图3描述的选择器120的另一输入。实际上，分配器114用机率分布的估计的先前提及的测度伴随每一字母符号。

[0054] 就图1A的熵编码设备的输出而言，该输出由经由VLC编码器102输出的第一位串流112及经由该多个熵编码器116输出的该多个第二位串流118组成。如以下进一步描述，可并行传输所有该位串流。或者，经由使用交错器128可将该位串流交错至共享位串流126中。图22至图24的实施例图标具有此位串流交错的实例。如图1A-图1C中进一步所示，PIPE编码器104本身可包含该PIPE编码器104自身的交错器130，以将该多个第二位串流118交错至共享PIPE编码位串流132中。此交错器130的可能性可源自以下关于图5至图13描述的实施例。位串流132及位串流112可以平行配置表示图1A的熵编码设备的输出。或者，另一交错器134可交错两个位串流，在此情况下交错器130及134将形成一个两阶段交错器128的两个阶段。

[0055] 如以上所述，再划分器100可执行再划分语法元素方式，亦即，再划分器100操作的源符号可为整个语法元素，或者换言的，再划分器100可以语法元素的单元操作。

[0056] 然而，根据替代性实施例，图1A的熵编码设备可包含分解器136，以将可剖析位串流138内的语法元素个别地分解成输入再划分器100的源符号序列106的一或更多源符号。特定言的，分解器136可经配置以经由将每一语法元素个别地分解成相应整数数目的源符号，来将语法元素的序列138转换成源符号的序列106。该整数数目可在语法元素之中改变。特定言的，该语法元素中的一些语法元素甚至可以经由分解器136保持不变，而将其它语法

元素用恰好两个或至少两个源符号分解。再划分器100可经配置以将该经分解语法元素的源符号中的一个源符号转发至源符号的第一子序列108，且将该经分解语法元素的源符号的另一个源符号转发至源符号的第二子序列110。如以上所提及，位串流138内语法元素可为不同类型，且分解器136可经配置以取决于语法元素的类型而执行个别分解。分解器136较佳地执行语法元素的个别分解，以使得存在为所有语法元素共享的自整数数目的源符号至相应语法元素的稍后在解码侧使用的预定唯一的逆向映射。

[0057] 举例而言，分解器136可经配置以将可剖析位串流138中的语法元素z分解成两个源符号x及y，以便z＝x+y、z＝x-y、z＝x·y或z＝x:y。经由此测度，再划分器100可将语法元素分解成两个分量，亦即源符号串流106的源符号，该两个分量中的一个分量(诸如，x)适合于依据压缩效率经VLC编码，且该两个分量中的另一个分量(诸如，y)不适合于经VLC编码，因此将该另一个分量传递至第二子串流110而非第一子串流108。由分解器136使用的分解不必为双射的。然而，如之前所提及，应存在逆向映射，该逆向映射实现可能的分解的语法元素的唯一检索，若分解不是双射的，则分解器136可在该可能分解之中选择。

[0058] 到现在为止，已针对不同语法元素的处理描述不同可能性。关于是否存在该语法元素或情况是视需要的。然而，本文所述的以下实施例集中于由分解器136根据以下原理分解的语法元素。

[0059] 如图1B中所示，分解器136经配置以分阶段分解可剖析位串流138中的某些语法元素z。两个或两个以上阶段可存在。该阶段是针对将语法元素z的值范围划分成两个或两个以上邻近子区间或图1C中所示的子范围。语法元素的值范围可具有两个无限的端点，仅一个或可能具有确定的端点。在图1C中，语法元素的值范围示例性地再划分成三个分区140_{1-3}。如图1B中所示，若语法元素大于或等于第一分区140_1的界限142，亦即，分隔分区140_1与140_2的上限，则将该语法元素减去第一分区140_1的界限limit1，且关于是否z甚至大于或等于第二分区140_2的界限144，亦即，分隔分区140_2与140_3的上限，来再次检查z。若z'大于或等于界限144，则将z'减去产生z⌋的第二分区140_2的界限limit2。在z小于limit1的第一情况下，将语法元素z简单地发送至再划分器100。在z介于limit1与limit2之间的情况下，将语法元素z作为元组(limit1，z')发送至再划分器100，其中z＝limit1+z'，且在z超过limit2的情况下，将语法元素z作为三字节(limit1，limit2-limit1，z')发送至再划分器100，其中z＝limit1+limit2+z'。第一(或唯一的)分量，亦即，z或limit1形成将经由再划分器100编码的第一源符号，若存在，则第二分量，亦即，z'或limit2-limit1形成将经由再划分器100编码的第二源符号，且若存在，则第三分量，亦即，z⌋形成将经由再划分器100编码的第三源符号。因此，根据图1B及图1C的实施例，将语法元素映像至1至3个源符号中的任一个，但对于差不多最大数目的源符号的一般化可易于源自以上描述，且亦将在下文中描述该替代物。

[0060] 在任何情况下，所有该不同分量或所得源符号根据以下实施例，用该实施例之中的编码替代物来编码。将该分量或源符号中的至少一个分量或源符号经由再划分器转发至PIPE编码器104，且最终将该分量或源符号中的另一个分量或源符号发送至VLC编码器102。

[0061] 以下更详细地概括特定有利的实施例。

[0062] 在描述以上熵编码设备的实施例的后，关于图2A描述熵解码设备的实施例。图2A的熵解码设备包含VLC解码器200及PIPE解码器202。VLC解码器200经配置以用编码的方式

自第一位串流206的码字组重建第一子序列204的源符号。第一位串流206等于图1A-图1C的位串流112，就图1A的子序列108而言，上述情况也适用于子序列204。PIPE解码器202经配置以重建以字母符号的序列的形式表示的源符号的第二子序列208，且PIPE解码器202包含多个熵解码器210、分配器212及选择器214。该多个熵解码器210经配置以将第二位串流216中的相应第二位串流216转换成字母符号的序列中的字母符号。分配器212经配置以基于字母符号的序列的先前重建的字母符号内含有的信息，将相应字母符号可假定的可能值之中的机率分布的估计测度分配予字母符号的序列中的每一字母符号，字母符号的该序列表示待重建的源符号的第二子序列208。为此，分配器212可串行连接于选择器214的输出与选择器214的输入之间，而选择器214的另外的输入具有熵解码器210的输出，熵解码器210的该输出分别连接至选择器214的该另外的输入。选择器214经配置以自该多个熵解码器210中的选定的熵解码器210检索字母符号的序列中的每一字母符号，该选择取决于分配予相应字母符号的测度。换言的，选择器214与分配器212一起经操作以在经由字母符号的序列中的先前字母符号内含有的测量信息获取的熵解码器210之中依次序检索经由熵解码器210获取的字母符号。甚至换言的，分配器212及选择器214能够重建不同字母符号之间的字母符号的原有次序。与预测下一个字母符号一起，分配器212能够决定相应字母符号的机率分布的估计的先前提及的测度，经由使用该测度，选择器214在熵解码器210之中选择检索此字母符号的实际值。如下文将更详细地描述，为了甚至更精确，PIPE解码器202可经配置以响应于顺序地请求字母符号的字母符号请求，重建以字母符号的序列的形式表示的源符号的子序列208，且分配器212可经配置以将相应字母符号可假定的可能值之中机率分布的估计的先前提及的测度分配予对于字母符号的序列中的字母符号的每一请求，字母符号的该序列表示待重建的源符号的第二子序列(208)。相应地，选择器214可经配置以针对对于字母符号的序列中的字母符号的每一请求，字母符号的该序列表示待重建的源符号的第二子序列(208)，自该多个熵解码器210中的选定的熵解码器210检索字母符号的序列中的相应字母符号，该选择取决于分配予对于相应字母符号的相应请求的测度。将关于图4更详细地概括一方面解码侧的请求与另一方面编码侧的数据流或编码之间的一致性。

[0063] 因为源符号的第一子序列214及源符号的第二子序列208通常形成源符号的一个共享序列210，所以图2A的熵解码设备可视需要包含复合器220，该复合器220经配置以再组合第一子序列204及第二子序列208，以获取源符号的共享序列218。源符号的此共享序列208产生图1A的序列106的再建。

[0064] 根据以上关于图1A-图1C提供的描述，第一子序列204及第二子序列208的源符号可为可剖析位串流的语法元素。在此情况下，复合器220可经配置以经由以由定义语法元素之中次序的一些剖析规则规定的次序交错经由第一序列204及第二子序列208到达的源符号，来重建语法元素的序列218的此可剖析位串流。特定言的，如以上所述，该语法元素可为不同类型，且复合器220可经配置以自VLC解码器200经由子串流204检索或请求第一组类型的语法元素且可经配置以自PIPE解码器202经由子串流208检索或请求第二类型的语法元素。相应地，每当刚刚提及的剖析规则指示第一组内类型的语法元素为成直线的下一个时，则复合器202将子序列204的实际源符号插入至共享序列218中，且自另外的子序列208插入。

[0065] 同样地，PIPE解码器202可包含连接于选择器214的输出与复合器220的输入之间

的去符号化器222。类似于以上关于图1A-图1C的描述，去符号化器222可视为在PIPE解码器202的外部，且甚至可将去符号化器222布置于复合器202的后，亦即，或者在复合器220的输出侧处。去符号化器222可经配置以在字母符号的部分序列的单元中将由选择器214输出的字母符号224的序列再映像至源符号亦即子序列208的语法元素中。类似于复合器220，去符号化器222了解字母符号的可能的部分序列的构造。特定言的，去符号化器222可分析最近自选择器214接收的字母符号，以确定关于该最近接收的字母符号是否产生与相应语法元素的相应值相关联的字母符号的有效的部分序列，或关于情况是否并非如此，以及接下来缺失哪个字母符号。甚至换言的，符号化器222在任何时间已知关于是否必须自选择器214接收另外的字母符号，以完成接收相应语法元素，且相应地，由选择器214输出的字母符号中的相应字母符号属于哪个语法元素。为此，去符号化器222可使用因不同类型的语法元素而不同的符号化(解)映像方案。类似地，分配器212了解将由选择器214自熵解码器210中的任一熵解码器210检索的当前字母符号与语法元素中的相应语法元素的关联，且分配器212可相应地设定此字母符号的机率分布的估计的以上提及的测度，亦即，取决于相关联的语法元素类型。甚至此外，分配器212可在属于当前字母符号的相同部分序列的不同字母符号之间区分，且分配器212可针对该字母符号以不同方式设定机率分布的估计的测度。以下更详细地描述在此方面的细节。如该处所描述的，分配器212可经配置以将上下文分配予字母符号。该分配可取决于语法元素类型及/或当前语法元素的字母符号的部分序列内的位置。分配器212一将上下文分配予将由选择器214自熵解码器210中的任一熵解码器210检索的当前字母符号，字母符号就可固有地具有与该上下文相关联的机率分布的估计的测度，因为每一上下文具有该上下文的与该上下文相关联的估计的测度。此外，可根据到现在为止已自熵解码器210检索的相应上下文的字母符号的实际的统计学，来调适上下文及该上下文的相关联的机率分布的估计的测度。以下更详细地呈现此方面的细节。

[0066] 类似于图1A-图1C的以上论述，有可能语法元素中子序列204及208的先前提及的源符号之间的对应不是一一对应的。相反地，可能已将语法元素分解成整数数目的源符号，其中该数目最终在语法元素之中变化，但在任何情况下，针对一个语法元素至少大于一个。如以上所注意到的，以下实施例集中于处理该种类的语法元素，且其它种类的语法元素甚至可以不存在。

[0067] 为处理刚刚提及的语法元素，图2A的熵解码设备可包含组合器224，该组合器224经配置以再进行由图1A的分解器136执行的分解。特定言的，组合器224可经配置以经由自相应整数数目的源符号个别地组合每一语法元素，而自序列218或在复合器220缺失的情况下子序列204及208的源符号组合语法元素的序列226，其中该整数数目的源符号中的源符号中的一个源符号属于第一子序列204，且相同语法元素的该整数数目的源符号中的源符号中的另一个源符号属于第二子序列208。经由此测度，可能已在编码器侧分解某些语法元素，以便将适合于VLC解码的分量与必须经由PIPE解码路径传递的剩余分量分隔。类似于以上论述，语法元素可为不同类型，且组合器224可经配置以取决于语法元素的类型而执行个别组合。特定言的，组合器224可经配置以经由在逻辑上或数学上组合相应语法元素的整数数目的源符号，来获取相应语法元素。举例而言，组合器224可经配置以针对每一语法元素将+、-、:或•应用于一个语法元素的第一源符号及第二源符号。

[0068] 然而，如以上所述，以下本文所述的实施例集中于经由根据图1B及图1C及关于图

1B及图1C描述的替代性实施例的分解器136分解的语法元素。图2A图标关于组合器224可如何发挥作用以自语法元素的源符号218重建该语法元素。

[0069] 如图2B中所示，组合器224经配置以分阶段自进入的源符号s_1至s_x组合该语法元素z，其中x为本实例中1至3中的任一个。两个或两个以上阶段可存在。如图2B中所示，组合器224初步将z设定为第一符号s_1且检查关于z是否等于第一limit1。若情况并非如此，则z已找到。否则，组合器224将源符号串流218的下一个源符号s_2添加至z且再次检查关于此z是否等于limit2。若z不等于limit2，则z已找到。否则，组合器224将源符号串流218的下一个源符号s_3添加至z，以获取呈最终形式的z。对差不多最大数目的源符号的一般化可易于源自以上描述，且亦将在下文中描述该替代物。

[0070] 在任何情况下，所有该不同分量或所得源符号根据以下实施例，用该实施例之中的编码替代物来编码。将该分量或源符号中的至少一个分量或源符号经由再划分器转发至PIPE编码器104，且最终将该分量或源符号中的另一个分量或源符号发送至VLC编码器102。

[0071] 以下更详细地概括特定有利的实施例。该细节集中于划分语法元素的值范围的有利可能性及可用以编码源符号的熵VLC编码及PIPE编码方案。

[0072] 此外，亦如以上关于图1A-图1C所述，图2A的熵解码设备可经配置以用交错位串流228的方式单独地或以交错形式接收第一位串流206以及该多个第二位串流216。在后者情况下，图2A的熵解码设备可包含去交错器230，该去交错器230经配置以将交错位串流228去交错，以获取一方面第一位串流206及另一方面该多个第二位串流216。类似于图1A-图1C的以上论述，可将去交错器230再分成两个阶段，亦即用于将交错位串流228去交错成两个部分的去交错器232，及用于将后者位串流234去交错以获取个别位串流216的去交错器236，该两个部分亦即一方面位串流206及另一方面第二位串流216的交错形式234。

[0073] 因此，图1A及图2A图标一方面熵编码设备及另一方面熵解码设备，该熵解码设备适合于解码经由图1A-图1C的熵编码设备获取的编码结果。关于另外的示图更详细地描述关于图1A及图2A和图2B中所示的许多元素的细节。相应地，就该细节可单独地实施于以上描述的实施例中而言，在以下描述中参阅该细节且该细节亦应视为个别地应用于图1A及图2A和图2B的实施例。仅关于交错器132及去交错器234，此处进行一些额外的通知。特定言的，位串流112及118的交错在必须将位串流多任务成一个通道以便传输的情况下可为有利的。在此情况下，可能有利的是，交错一方面VLC位串流112及另一方面PIPE编码位串流118，以便遵照某些要满足的条件，诸如遵照一些最大解码延迟。甚至换言的，可能有必要的是，一方面在解码侧分别可检索语法元素及源符号的时间之间的相对时间位移及另一方面根据该语法元素及源符号在可剖析位串流中的位置的时间的相对位移不超过某一最大延迟。以下描述用于解决此问题的许多替代性实施例。该可能性中的一个可能性涉及熵编码器116及熵解码器210，该熵编码器116为可变长度编码器类型，该可变长度编码器类型经配置以将字母符号序列映像至码字组，该熵解码器210执行逆向映射。VLC位串流112及PIPE位串流118的码字组可(但不必)经选择以使得没有该位串流中的任一位串流的码字组为其它位串流中的任一位串流的任何码字组的前缀，以便保持唯一可在解码器端决定码字组界限。在任何情况下，交错器128可经配置以取决于选择器120转发至该多个熵编码器116的字母符号的序列124中的字母符号产生将在相应熵编码器116处映像至相应码字组的新的字母符号序列的开始、且第二子串流108的新的源符号经由VLC编码器102映像的一个次序，而按

一顺序次序分别为第一位串流112及第二位串流118内的码字组保留且缓冲码字组项的序列。换言的，交错器128将位串流112的码字组按源符号的次序插入至共享位串流126中，自该共享位串流126，已经由VLC编码分别按该源符号在子串流108及源符号串流106内的次序获取该源符号。将由熵编码器116输出的码字组插入至VLC位串流112的码字组中的连续码字组之间的共享位串流126中。由于分别经由分配器114及选择器120的字母符号的PIPE编码分类，熵编码器116的码字组中的每一码字组具有在该处编码的子串流110的不同源符号的字母符号。分别经由在每一码字组中编码的第一字母符号(亦即，时间上最久的字母符号)决定彼此之中且相对于位串流112的VLC码字组的共享位串流126内PIPE编码位串流118的码字组的位置。编码成字母符号串流124中位串流118的码字组的该初级字母符号的次序决定彼此之中共享位串流126内位串流118的码字组的次序；相对于位串流112的VLC码字组，编码成位串流118的码字组的该初级字母符号所属于的源符号决定位串流118中的任一个位串流118的相应码字组将定位于位串流112的哪些连续的码字组之间。特定言的，位串流118中的任一个位串流118的相应码字组将定位于其间的连续VLC码字组为根据未再划分源符号串流106的原有次序，将子串流110的源符号定位于其间的连续VLC码字组，编码成位串流118的相应码字组的相应初级字母符号属于该未再划分源符号串流106。交错器128可经配置以按顺序的次序移除输入先前提及的码字组项中的码字组，以获取交错码字组的共享位串流126。如以上已描述的，熵编码器116可经配置以将该熵编码器116的码字组顺序地输入针对相应熵编码器116保留的码字组项中，且选择器120可经配置以按使第一子串流108及第二子串流110的源符号在源符号的序列106内交错的次序维持住的一个次序，来转发表示第二子串流110的源符号的字母符号。

[0074] 可提供额外的测度，以应对熵编码器116中的某些熵编码器116如此很少地经选择以致于花费长时间来获取极少使用的熵编码器116内有效码字组的情形。以下更详细地描述该测度的实例。特定言的，在此情况下，交错器128与熵编码器116一起可经配置以分别以一定方式清除到现在为止收集的该交错器128及熵编码器116的字母符号及已输入先前提及的码字组项中的码字组，该方式使得可在解码侧预测或仿真此清除步骤的时间。

[0075] 在解码侧，去交错器230可在相反意义上作用：根据先前提及的剖析方案，每当待解码的下一个源符号为VLC编码符号时，共享位串流228内的当前码字组视为VLC码字组且在位串流206内转发至VLC解码器200。另一方面，每当属于子串流208的PIPE编码符号中的任一PIPE编码符号的字母符号中的任一字母符号为初级字母符号，亦即，必须经由相应熵解码器210将位串流216中的相应位串流216的码字组重新映像至相应字母符号序列时，将共享位串流228的当前码字组视为PIPE编码码字组且转发至相应熵解码器210。下一个码字组界限的侦测，亦即，下一个码字组自刚刚已分别转发至解码器200及202中的任一者的码字组的末端延伸至该下一个码字组在入端口交错位串流228内的末端的侦测可延期且可在解码器200及202为根据以上概括的规则的此下一个码字组的专用接收者的知识的下执行：基于此知识，已知由接收者解码器使用的密码本且可侦测相应码字组。另一方面，若密码本将经设计以使得码字组界限将可侦测，而不具有关于200及202之中接收者解码器的先验知识，则可平行执行码字组分隔。在任何情况下，归因于交错，可在解码器处以熵解码形式(亦即，作为源符号)按该源符号的正确次序以合理的延迟获得源符号。

[0076] 在描述熵编码设备及相应熵解码设备的以上实施例的后，接下来描述以上提及的

PIPE编码器及PIPE解码器的更详细的实施例。

[0077] 在图3中图示根据实施例的PIPE编码器。该PIPE编码器可用作图1A中的PIPE编码器。PIPE编码器将源符号1的串流无损地转换成两个或两个以上部分位串流12的集合。每一源符号1可与一或更多类别或类型的集合中的类别或类型相关联。例如，类别可指定源符号的类型。在混合视讯编码的上下文中，单独的类别可与宏区块编码模式、分块编码模式、参阅图像标示、移动向量差异、再划分旗标、编码区块旗标、量化参数、变换系数位准等相关联。在诸如音讯、语音、本文、文献或一般数据编码的其它应用领域中，源符号的不同分类是可能的。

[0078] 一般而言，每一源符号可取有限值或值的可数无限集，其中可能源符号值的设定可因不同源符号类别而不同。为降低编码及解码算法的复杂性且为允许不同源符号及源符号类别的一般的编码及解码设计，将源符号1转换成二元决策的有次序的集合，且随后经由简单的二进制编码算法处理该二元决策。因此，二进制编码器2将每一源符号1的值以双射方式映像至二进制3的序列(或字符串)上。二进制3的序列表示有次序的二元决策的集合。每一二进制3或二元决策可取两个值的集合中的一个值，例如，值0及1中的一者。二进制化方案可因不同源符号类别而不同。特定的源符号类别的二进制化方案可取决于可能的源符号值的集合及/或该特定类别的源符号的其它性质。表1图标可数无限集合的三个示例性二进制化方案。可数无限集合的二进制化方案亦可应用于符号值的有限集合。特定言的，对于符号值的有限集合而言，低效率(由二进制的未使用的序列产生)可忽略，但该二进制化方案的普遍性提供依据复杂性及内存要求的优点。对于符号值的小的有限集合而言，通常较佳(依据编码效率)的是使二进制化方案适应可能符号值的数目。表2图标8个值有限集合的三个示例性二进制化方案。经由以二进制序列的有限集合表示无冗余度码(且可能将二进制序列重新排序)的方式修改二进制的一些序列，有限集合的二进制化方案可源自于可数无限集合的通用的二进制化方案。例如，经由修改通用一元二进制化(参见表1)的源符号7的二进制序列，来产生表2中的截断式一元二进制化方案。经由修改通用的指数哥伦布0阶二进制化(参见表1)的源符号7的二进制序列且经由重新排序该二进制序列(将符号7的截断式二进制序列分配予符号1)，来产生表2中的截断式且重新排序的指数哥伦布0阶二进制化。对于符号的有限集合而言，亦可能使用非系统/非通用二进制化方案，如表2的最后一行中所例示的。

[0079] 表1：可数无限集合(或较大的有限集合)的二进制化实例。

[0080]

符号值	一元二进制化	指数哥伦布0阶二进制化	指数哥伦布1阶二进制化
0	1	1	10
1	01	010	11
2	001	011	0100
3	0001	0010 0	0101
4	0000 1	0010 1	0110
5	0000 01	0011 0	0111
6	0000 001	0011 1	0010 00
7	0000 0001	0001 000	0010 01
...	...	...	...

[0081] 表2:有限集合的二进制化实例。

[0082]

符号值	截断式一元二进制化	截断式且重新排序的指数哥伦布0阶二进制化	非系统二进制化
0	1	1	000
1	01	000	001
2	001	010	01

[0083]

3	0001	011	1000
4	0000 1	0010 0	1001
5	0000 01	0010 1	1010
6	0000 001	0011 0	1011 0
7	0000 000	0011 1	1011 1

[0084] 将经由二进制编码器2产生的二进制的序列的每一二进制3按顺序的次序馈送至参数分配器4中。该参数分配器将一或更多参数的集合分配予每一二进制3且输出具有参数5的相关联集合的二进制。在编码器与解码器处以恰好相同的方式决定参数的集合。参数的集合可由以下参数中的一或更多参数组成:

[0085] -针对当前二进制的两个可能二进制值中的一个二进制值的机率的估计的测度,

[0086] -针对当前二进制的更小机率或更大机率的二进制值的机率的估计的测度,

[0087] -指定对于两个可能二进制值中的哪个表示当前二进制的更小机率或更大机率的二进制值的估计的识别符,

[0088] -相关联源符号的类别,

[0089] -相关联源符号的重要性的测度,

[0090] -相关联符号的位置的测度(例如,时间的、空间的或容积数据集),

[0091] -指定二进制或相关联源符号的信道码保护的识别符,

[0092] -指定二进制或相关联源符号的加密方案的识别符,

[0093] -指定相关联符号的级别的识别符,

[0094] -相关联源符号的二进制的序列中的二进制数目。

[0095] 在实施例中,参数分配器4使每一二进制3、5与针对当前二进制的两个可能二进制值中的一个二进制值的机率的估计的测度相关联。参数分配器4使每一二进制3、5与针对当前二进制的更小机率或更大机率的二进制值的机率的估计的测度及指定对于两个可能二进制值中的哪个表示当前二进制的更小机率或更大机率的二进制值的估计的识别符相关联。应注意,更小机率或更大机率的二进制值的机率及指定两个可能二进制值中的哪个表示该更小机率或更大机率的二进制值的识别符为该两个可能二进制值中的一个二进制值的机率的等效测度。

[0096] 在另一实施例中,参数分配器4使每一二进制3、5与针对当前二进制的两个可能二进制值中的一个二进制值的机率的估计的测度及一或更多另外的参数(该一或更多另外的参数可为以上列出参数中的一或更多参数)相关联。在另一实施例中,参数分配器4使每一二进制3、5与针对当前二进制的更小机率或更大机率的二进制值的机率的估计的测度、指定对于两个可能二进制值中的哪个表示当前二进制的更小机率或更大机率的二进制值的估计的识别符及一或更多另外的参数(该一或更多另外的参数可为以上列出参数中的一或更多参数)相关联。

[0097] 在实施例中,参数分配器4基于一或更多已编码符号的集合,决定以上提及的机率测度中的一或更多机率测度(针对当前二进制的两个可能二进制值中的一个二进制值的机率的估计的测度、针对当前二进制的更小机率或更大机率的二进制值的机率的估计的测度、指定对于两个可能二进制值中的哪个表示当前二进制的更小机率或更大机率的二进制值的估计的识别符)。用于决定机率测度的编码符号可包括相同符号类别的一或更多已编码符号、对应于邻近空间及/或时间位置(相对于与当前源符号相关联的数据集)的数据集(诸如,取样的区块或群组)的相同符号类别的一或更多已编码符号或对应于相同及/或邻近空间及/或时间位置(相对于与当前源符号相关联的数据集)的数据集的不同符号类别的一或更多已编码符号。

[0098] 将具有参数5的相关联集合的每一二进制馈送至二进制缓冲选择器6中,参数5的该相关联集合为参数分配器4的输出。二进制缓冲选择器6可能基于输入二进制值及相关联参数5修改输入二进制5的值,且二进制缓冲选择器6将具有可能经修改值的输出二进制7馈送至两个或两个以上二进制缓冲器8中的一个二进制缓冲器8中。基于输入二进制5的值及/或相关联参数5的值,决定二进制缓冲器8,输出二进制7发送至该二进制缓冲器8。

[0099] 在实施例中,二进制缓冲选择器6不修改二进制的值,亦即,输出二进制7具有始终与输入二进制5相同的值。

[0100] 在另一实施例中,二进制缓冲选择器6基于输入二进制值5及当前二进制的两个可能二进制值中的一个二进制值的机率的估计的相关联测度,决定输出二进制值7。在实施例中,若当前二进制的两个可能二进制值中的一个二进制值的机率的测度小于(或小于或等于)特定临界值,则将输出二进制值7设定成等于输入二进制值5;若当前二进制的两个可能

二进制值中的一个二进制值的机率的测度大于或等于(或大于)特定临界值,则修改输出二进制值7(亦即,将输出二进制值7设定成与输入二进制值相反)。在另一实施例中,若当前二进制的两个可能二进制值中的一个二进制值的机率的测度大于(或大于或等于)特定临界值,则将输出二进制值7设定成等于输入二进制值5;若当前二进制的两个可能二进制值中的一个二进制值的机率的测度小于或等于(或小于)特定临界值,则修改输出二进制值7(亦即,将输出二进制值7设定成与输入二进制值相反)。在实施例中,该临界值对应于两个可能二进制值的估计机率的值0.5。

[0101] 二进制缓冲选择器6可基于输入二进制值5及指定对于两个可能二进制值中的哪个表示当前二进制的更小机率或更大机率的二进制值的估计的相关联识别符,决定输出二进制值7。在实施例中,若识别符指定两个可能二进制值中的第一二进制值表示当前二进制的更小机率(或更大机率)的二进制值,则将输出二进制值7设定成等于输入二进制值5,且若识别符指定两个可能二进制值中的第二二进制值表示当前二进制的更小机率(或更大机率)的二进制值,则修改输出二进制值7(亦即,将输出二进制值7设定成与输入二进制值相反)。

[0102] 二进制缓冲选择器6可基于当前二进制的两个可能二进制值中的一个二进制值的机率的估计的相关联测度,决定二进制缓冲器8,输出二进制7发送至该二进制缓冲器8。在实施例中,两个可能二进制值中的一个二进制值的机率的估计的测度的可能值的集合为有限的,且二进制缓冲选择器6含有表,该表恰好使一个二进制缓冲器8与该两个可能二进制值中的一个二进制值的机率的估计的每一可能值相关联,其中该两个可能二进制值中的一个二进制值的机率的估计的测度的不同值可与相同二进制缓冲器8相关联。在另一实施例中,将两个可能二进制值中的一个二进制值的机率的估计的测度的可能值范围分割成若干区间,二进制缓冲选择器6决定该两个可能二进制值中的一个二进制值的机率的估计的当前测度的区间标示,且二进制缓冲选择器6含有表,该表恰好使一个二进制缓冲器8与该区间标示的每一可能值相关联,其中该区间标示的不同值可与相同二进制缓冲器8相关联。在较佳实施例中,将具有两个可能二进制值中的一个二进制值的机率的估计的相反测度(相反测度为表示机率估计P及1-P的相反测度)的输入二进制5馈送至相同二进制缓冲器8中。在另一较佳实施例中,当前二进制的两个可能二进制值中的一个二进制值的机率的估计的测度与特定二进制缓冲器的关联随时间调适,(例如)以确保所产生的部分位串流具有类似位率。

[0103] 二进制缓冲选择器6可基于当前二进制的更小机率或更大机率的二进制值的机率的估计的相关联测度,决定二进制缓冲器8,输出二进制7发送至该二进制缓冲器8。在较佳实施例中,更小机率或更大机率的二进制值的机率的估计的测度的可能值的集合为有限的,且二进制缓冲选择器6含有表,该表恰好使一个二进制缓冲器8与更小机率或更大机率的二进制值的机率的估计的每一可能值相关联,其中该更小机率或更大机率的二进制值的机率的估计的测度的不同值可与相同二进制缓冲器8相关联。在另一实施例中,将更小机率或更大机率的二进制值的机率的估计的测度的可能值范围分割成若干区间,二进制缓冲选择器6决定该更小机率或更大机率的二进制值的机率的估计的当前测度的区间标示,且二进制缓冲选择器6含有表,该表恰好使一个二进制缓冲器8与该区间标示的每一可能值相关联,其中该区间标示的不同值可与相同二进制缓冲器8相关联。在另一实施例中,当前二进

制的更小机率或更大机率的二进制值的机率的估计的测度与特定二进制缓冲器的关联随时间调适，(例如)以确保所产生的部分位串流具有类似位率。

[0104] 两个或两个以上二进制缓冲器8中的每一二进制缓冲器8恰好与一个二进制编码器10连接，且每一二进制编码器仅与一个二进制缓冲器8连接。每一二进制编码器10自相关联二进制缓冲器8读取二进制且将二进制9的序列转换成码字组11，该码字组11表示位的序列。二进制缓冲器8表示先进先出缓冲器；随后(按顺序的次序)被馈送至二进制缓冲器8中的二进制并不在较早地(按顺序的次序)被馈送至该二进制缓冲器中的二进制之前编码。将为特定二进制编码器10的输出的码字组11写入至特定部分位串流12。整体编码算法将源符号1转换成两个或两个以上部分位串流12，其中部分位串流的数目等于二进制缓冲器及二进制编码器的数目。在较佳实施例中，二进制编码器10将可变量目的二进制9转换成可变量目的位的码字组11。以上及以下概括的PIPE编码实施例的一个优点为，可平行(例如，对于机率测度的不同群组而言)执行二进制的编码，该优点减少若干实施的处理时间。

[0105] PIPE编码的另一优点为，可针对参数5的不同集合具体设计经由二进制编码器10执行的二进制编码。特定言的，可针对经估计机率的不同群组最佳化(依据编码效率及/或复杂性)二进制编码及编码。一方面，此举允许降低相对于具有类似编码效率的算术编码算法的编码/解码复杂性。另一方面，此举允许改良相对于具有类似编码/解码复杂性的VLC编码算法的编码效率。在实施例中，二进制编码器10针对当前二进制的两个可能二进制值5中的一个二进制值5的机率的估计的测度的不同群组实施不同编码算法(亦即，将二进制序列映射至码字组上)。在另一实施例中，二进制编码器10针对当前二进制的更小机率或更大机率的二进制值的机率的估计的测度的不同群组实施不同编码算法。在另一实施例中，二进制编码器10针对不同通道保护码实施不同编码算法。在另一实施例中，二进制编码器10针对不同加密方案实施不同编码算法。在另一实施例中，二进制编码器10针对当前二进制的两个可能二进制值5中的一个二进制值5的机率的估计的测度的通道保护码及群组的不同组合实施不同编码算法。在另一实施例中，二进制编码器10针对当前二进制的更小机率或更大机率的二进制值5的机率的估计的测度的通道保护码及群组的不同组合实施不同编码算法。在另一实施例中，二进制编码器10针对当前二进制的两个可能二进制值5中的一个二进制值5的机率的估计的测度的加密方案及群组的不同组合实施不同编码算法。在另一较佳实施例中，二进制编码器10针对当前二进制的更小机率或更大机率的二进制值5的机率的估计的测度的加密方案及群组的不同组合实施不同编码算法。

[0106] 二进制编码器10或一或更多二进制编码器可表示二进制算术编码引擎。在另一实施例中，一或更多二进制编码器表示二进制算术编码引擎，其中经由使用表查找来实现自给定二进制缓冲器的代表性的LPS/LPB机率p_{LPS}映射至相应的码区间宽度R_{LPS}，亦即，二进制算术编码引擎的内部状态的区间再划分识别(例如)码区间的下限，该区间再划分由当前区间宽度R及当前区间偏差L定义。在另一较佳实施例中，对于与给定二进制缓冲器相关联的每一基于表的二进制算术编码引擎而言，K代表的区间宽度值$\{Q_0,\ldots,Q_{K-1}\}$用于表示具有K的选择的R_{LPS}及取决于二进制缓冲器的代表性的区间宽度值$\{Q_0,\ldots,Q_{K-1}\}$。对于K>1的选择而言，二进制的算术编码可涉及将当前区间宽度R映射至具有对于K＝1的选择而言，在仅给出一个代表性的区间宽度值Q_0的情况下，可将此值Q_0选作二的幂，以允许解码在单一再归一化周期内输入相应二进制缓冲器的多个MPS/MPB值。可单独地传输、包化或储存每一算术编

码引擎的所得码字组,或出于如在下文中描述的传输或储存的目的,可将该码字组交错。

[0107] 亦即,二进制算术编码引擎10可执行在该二进制算术编码引擎10的二进制缓冲器8中编码二进制的以下步骤:

[0108] 1.自二进制缓冲器接收valLPS二进制(再呼叫:已选择此处考虑的相应二进制算术编码引擎10接收「二进制」,因为(或,换言的,「二进制」与相应二进制算术编码引擎10相关联)诸如p_state[bin]的机率分布估计与彼二进制算术编码引擎10相关联)

[0109] 2.R的量化:

[0110] q_index=Qtab[R>>q](或一些其它形式的量化)

[0111] 3.R_{LPS}及R的决定:

[0112] R_{LPS}=Rtab[q_index](注意p_state在此处并未提及,因为p_state固定用于所考虑的二进制算术编码引擎10,亦即p_state[encoder],且Rtab在该处已储存p[p_state[encoder]]·Q[q_index]的预先计算值

[0113] R=R-R_{LPS}[亦即,初步预先更新R,似乎「二进制」为MPS]

[0114] 4.新的部分区间的计算:

[0115] 若(二进制=1-valMPS),则

[0116] L←L+R

[0117] R←R_{LPS}

[0118] 5.L及R的再归一化,写入位,

[0119] 其中

[0120] q_index描述由Qtab读取的量化值的标示,

[0121] p_state描述当前状态(固定用于二进制算术编码引擎10),

[0122] R_{LPS}描述对应于LPS的区间宽度及

[0123] valMPS描述对应于MPS的位值。

[0124] 相应地,二进制算术解码引擎22可执行将二进制输出解码成二进制缓冲器20的以下步骤:

[0125] 1.接收对于二进制的请求(再呼叫:已选择此处考虑的相应二进制算术解码引擎22解码「二进制」,因为(或,换言的,「二进制」与相应二进制算术解码引擎22相关联)诸如p_state[bin]的机率分布估计与彼二进制算术解码引擎22相关联)

[0126] 2.R的量化:

[0127] q_index=Qtab[R>>q](或一些其它形式的量化)

[0128] 3.R_{LPS}及R的决定:

[0129] R_{LPS}=Rtab[q_index](注意p_state在此处并未提及,因为p_state固定用于所考虑的二进制算术解码引擎22,亦即p_state[encoder],且Rtab在该处已储存p[p_state[encoder]]·Q[q_index]的预先计算值

[0130] R=R-R_{LPS}[亦即,初步预先更新R,似乎「二进制」为MPS]

[0131] 4.取决于部分区间的位置的二进制的决定:

[0132] 若(V≧R)则

[0133] 二进制←1-valMPS(将二进制解码为LPS;二进制缓冲选择器18将经由使用此二进制信息及valMPS来获取实际的二进制值)

[0134] V←V-R

[0135] $R \leftarrow R_{LPS}$

[0136] 否则

[0137] 二进制←valMPS(将二进制解码为MPS;二进制缓冲选择器18将经由使用此二进制信息及valMPS来获取实际的二进制值)

[0138] 5.R的再归一化,读出一个位且更新V,

[0139] 其中

[0140] q_index描述由Qtab读取的量化值的标示,

[0141] p_state描述当前状态(固定用于二进制算术解码引擎22),

[0142] R_{LPS}描述对应于LPS的区间宽度,

[0143] valMPS描述对应于MPS的位值,及

[0144] V描述来自当前部分区间的内部的值。

[0145] 二进制编码器10或一或更多二进制编码器可表示熵编码器,该熵编码器将输入二进制9的序列直接映像至码字组10上。该映像可有效率地实施且不需要复杂的算术编码引擎。将码字组逆映射至二进制的序列上(如在解码器中执行的)应为唯一的,以保证完美解码输入序列,但将二进制序列9映像至码字组10上未必需要是唯一的,亦即,有可能可将二进制的特定序列映射至码字组的多于一个序列上。在实施例中,将输入二进制9的序列映射至码字组10上为双射的。在另一较佳实施例中,二进制编码器10或一或更多二进制编码器表示熵编码器,该熵编码器将输入二进制9的可变长度序列直接映像至可变长度码字组10上。在实施例中,输出码字组表示无冗余度码,诸如,一般的霍夫曼码或典型的霍夫曼码。

[0146] 将二进制序列双射映射至无冗余度码的两个实例图标于表3中。在另一实施例中,输出码字组表示适合于错误侦测及错误恢复的冗余码。在另一较佳实施例中,输出码字组表示适合于加密源符号的加密码。

[0147] 表3:在二进制序列与码字组之间映像的实例。

[0148]

二进制的序列 (二进制次序为自左至右)	码字组 (位次序为自左至右)
0000 0000	1
0000 0001	0000

[0149]

0000 001	0001
0000 01	0010
0000 1	0011
0001	0100
001	0101
01	0110
1	0111

[0150]

二进制的序列 （二进制次序为自左至右）	码字组 （位次序为自左至右）
000	10
01	11
001	010
11	011
1000 0	0001
1001	0010
1010	0011
1000 1	0000 0
1011	0000 1

[0151] 在另一实施例中，二进制编码器10或一或更多二进制编码器表示熵编码器，该熵编码器将输入二进制9的可变长度序列直接映像至固定长度码字组10上。在另一实施例中，二进制编码器10或一或更多二进制编码器表示熵编码器，该熵编码器将输入二进制9的固定长度序列直接映像至可变长度码字组10上。

[0152] 根据实施例的PIPE解码器图示于图4中。解码器基本上执行图3的编码器的逆向操作，以便自两个或两个以上部分位串流24的集合解码源符号27的(先前编码的)序列。解码器包括两个不同处理流程：数据请求流及数据流，该数据请求流复制编码器的数据流，该数据流表示编码器数据流的逆向。在图4的说明中，虚线箭头表示资料请求流，而实线箭头表示资料流。解码器的构建块基本上复制编码器的构建块，但实施逆向操作。

[0153] 源符号的解码经由请求新的解码源符号13来触发，该新的解码源符号13被发送至二进制编码器14。在实施例中，对于新的解码源符号13的每一请求与一或更多类别的集合中的类别相关联。与对于源符号的请求相关联的类别同与编码期间相应的源符号相关联的类别相同。

[0154] 二进制编码器14将对于源符号13的请求映像至对于发送至参数分配器16的二进制的一或更多请求中。作为对于经由二进制编码器14发送至参数分配器16的二进制的请求的最终响应，二进制编码器14自二进制缓冲选择器18接收经解码二进制26。二进制编码器14将所接收的经解码二进制26的序列与所请求的源符号的特定二进制化方案的二进制序列比较，且若所接收的经解码二进制26的序列与源符号的二进制化匹配，则二进制编码器清空该二进制编码器的二进制缓冲器且输出解码源符号作为对新的解码符号的请求的最终响应。若已接收的经解码二进制的序列与所请求的源符号的特定二进制化方案的二进制序列中的任一二进制序列不匹配，则二进制编码器将对于二进制的另一请求发送至参数分配器，直至经解码二进制的序列与所请求的源符号的二进制化方案的二进制序列中的一个二进制序列匹配为止。对于源符号的每一请求而言，解码器使用用于编码相应的源符号的相同二进制化方案。二进制化方案可因不同源符号类别而不同。特定的源符号类别的二进制化方案可取决于可能的源符号值的集合及/或该特定类别的源符号的其它性质。

[0155] 参数分配器将一或更多参数的集合分配予对于二进制的每一请求，且参数分配器

将对于具有参数的相关联集合的二进制的请求发送至二进制缓冲选择器。经由参数分配器分配予所请求的二进制的参数的集合与在编码期间分配予相应二进制的参数的集合相同。参数的集合可由在编码器描述中提及的一或更多参数组成。

[0156] 参数分配器16可使对于二进制的每一请求与当前所请求的二进制的两个可能二进制值中的一个二进制值的机率的估计的测度相关联。特定言的，参数分配器16可使对于二进制的每一请求与针对当前所请求的二进制的更小机率或更大机率的二进制值的机率的估计的测度及指定对于两个可能二进制值中的哪个表示当前所请求的二进制的更小机率或更大机率的二进制值的估计的识别符相关联。

[0157] 在另一实施例中，参数分配器16可使对于二进制15、17的每一请求与当前所请求的二进制的两个可能二进制值中的一个二进制值的机率的估计的测度及一或更多另外的参数相关联。在另一较佳实施例中，参数分配器16使对于二进制15、17的每一请求与针对当前所请求的二进制的更小机率或更大机率的二进制值的机率的估计的测度、指定对于两个可能二进制值中的哪个表示当前所请求的二进制的更小机率或更大机率的二进制值的估计的识别符及一或更多另外的参数(该一或更多另外的参数可为以上列出参数中的一或更多参数)相关联。

[0158] 在实施例中，参数分配器16基于一或更多已解码符号的集合，决定以上提及的机率测度中的一或更多机率测度(针对当前所请求的二进制的两个可能二进制值中的一个二进制值的机率的估计的测度、针对当前所请求的二进制的更小机率或更大机率的二进制值的机率的估计的测度、指定对于两个可能二进制值中的哪个表示当前所请求的二进制的更小机率或更大机率的二进制值的估计的识别符)。对于二进制的特定请求的机率测度的决定复制相应二进制的编码器处的程序。用于决定机率测度的解码符号可包括相同符号类别的一或更多已解码符号、对应于邻近空间及/或时间位置(相对于与源符号的当前请求相关联的数据集)的数据集(诸如，取样的区块或群组)的相同符号类别的一或更多已解码符号或对应于相同及/或邻近空间及/或时间位置(相对于与源符号的当前请求相关联的数据集)的数据集的不同符号类别的一或更多已解码符号。

[0159] 将对于具有参数17的相关联集合的二进制的每一请求馈送至二进制缓冲选择器18中，参数17的该相关联集合为参数分配器16的输出。基于参数17的相关联集合，二进制缓冲选择器18将对于二进制19的请求发送至两个或两个以上二进制缓冲器20中的一个二进制缓冲器，且二进制缓冲选择器18自选定的二进制缓冲器20接收经解码二进制25。可能修改解码输入二进制25，且将具有可能的修改值的解码输出二进制26发送至二进制编码器14作为对具有参数17的相关联集合的二进制的请求的最终回应。

[0160] 以与编码器侧的二进制缓冲选择器的输出二进制被发送至的二进制缓冲器相同的方式选择二进制缓冲器20，对于二进制的请求被转发至该二进制缓冲器20。

[0161] 在实施例中，二进制缓冲选择器18基于当前所请求的二进制的两个可能二进制值中的一个二进制值的机率的估计的相关联测度，决定二进制缓冲器20，对于二进制19的请求被发送至该二进制缓冲器20。在实施例中，两个可能二进制值中的一个二进制值的机率的估计的测度的可能值的集合为有限的，且二进制缓冲选择器18含有表，该表恰好使一个二进制缓冲器20与该两个可能二进制值中的一个二进制值的机率的估计的每一可能值相关联，其中该两个可能二进制值中的一个二进制值的机率的估计的测度的不同值可与相同

二进制缓冲器20相关联。在另一实施例中，将两个可能二进制值中的一个二进制值的机率的估计的测度的可能值范围分割成若干区间，二进制缓冲选择器18决定该两个可能二进制值中的一个二进制值的机率的估计的当前测度的区间标示，且二进制缓冲选择器18含有表，该表恰好使一个二进制缓冲器20与该区间标示的每一可能值相关联，其中该区间标示的不同值可与相同二进制缓冲器20相关联。可将对于具有两个可能二进制值中的一个二进制值的机率的估计的相反测度(相反测度为表示机率估计P及1-P的相反测度)的二进制17的请求转发至相同二进制缓冲器20中。此外，当前二进制请求的两个可能二进制值中的一个二进制值的机率的估计的测度与特定二进制缓冲器的关联可随时间调适。

[0162] 二进制缓冲选择器18可基于当前所请求的二进制的更小机率或更大机率的二进制值的机率的估计的相关联测度，决定二进制缓冲器20，对于二进制19的请求被发送至该二进制缓冲器20。在实施例中，更小机率或更大机率的二进制值的机率的估计的测度的可能值的集合为有限的，且二进制缓冲选择器18含有表，该表恰好使一个二进制缓冲器20与更小机率或更大机率的二进制值的机率的估计的每一可能值相关联，其中该更小机率或更大机率的二进制值的机率的估计的测度的不同值可与相同二进制缓冲器20相关联。在实施例中，更小机率或更大机率的二进制值的机率的估计的测度的可能值范围分割成若干区间，二进制缓冲选择器18决定该更小机率或更大机率的二进制值的机率的估计的当前测度的区间标示，且二进制缓冲选择器18含有表，该表恰好使一个二进制缓冲器20与该区间标示的每一可能值相关联，其中该区间标示的不同值可与相同二进制缓冲器20相关联。在另一较佳实施例中，当前二进制请求的更小机率或更大机率的二进制值的机率的估计的测度与特定二进制缓冲器的关联随时间调适。

[0163] 在自选定的二进制缓冲器20接收经解码二进制25的后，二进制缓冲选择器18可能修改输入二进制25且将具有可能的修改值的输出二进制26发送至二进制编码器14。二进制缓冲选择器18的输入/输出二进制映像与编码器侧的二进制缓冲选择器的输入/输出二进制映射逆向。

[0164] 二进制缓冲选择器18可经配置以不修改二进制的值，亦即，输出二进制26具有始终与输入二进制25相同的值。

[0165] 二进制缓冲选择器18可基于输入二进制值25及与对于二进制17的请求相关联的当前所请求的二进制的两个可能二进制值中的一个二进制值的机率的估计的测度，决定输出二进制值26。在实施例中，若当前二进制请求的两个可能二进制值中的一个二进制值的机率的测度小于(或小于或等于)特定临界值，则将输出二进制值26设定成等于输入二进制值25；若当前二进制请求的两个可能二进制值中的一个二进制值的机率的测度大于或等于(或大于)特定临界值，则修改输出二进制值26(亦即，将输出二进制值26设定成与输入二进制值相反)。在另一实施例中，若当前二进制请求的两个可能二进制值中的一个二进制值的机率的测度大于(或大于或等于)特定临界值，则将输出二进制值26设定成等于输入二进制值25；若当前二进制请求的两个可能二进制值中的一个二进制值的机率的测度小于或等于(或小于)特定临界值，则修改输出二进制值26(亦即，将输出二进制值26设定成与输入二进制值相反)。在较佳实施例中，该临界值对应于两个可能二进制值的估计机率的值0.5。

[0166] 在另一实施例中，二进制缓冲选择器18基于输入二进制值25及指定对于两个可能二进制值中的哪个表示当前二进制请求的更小机率或更大机率的二进制值的估计的与对

于二进制17的请求相关联的识别符，决定输出二进制值26。在较佳实施例中，若识别符指定两个可能二进制值中的第一二进制值表示当前二进制请求的更小机率(或更大机率)的二进制值，则将输出二进制值26设定成等于输入二进制值25，且若识别符指定两个可能二进制值中的第二二进制值表示当前二进制请求的更小机率(或更大机率)的二进制值，则修改输出二进制值26(亦即，将输出二进制值26设定成与输入二进制值相反)。

[0167] 如以上所述，二进制缓冲选择器将对于二进制19的请求发送至两个或两个以上二进制缓冲器20中的一个二进制缓冲器20。二进制缓冲器20表示先进先出缓冲器，自经连接二进制解码器22以经解码二进制21的序列馈送给该先进先出缓冲器。作为对自二进制缓冲选择器18发送至二进制缓冲器20的二进制19的请求的回应，二进制缓冲器20移除被首先馈送至二进制缓冲器20中的该二进制缓冲器20的内容的二进制且将该二进制发送至二进制缓冲选择器18。较早地移除被较早地发送至二进制缓冲器20的二进制且将该二进制发送至二进制缓冲选择器18。

[0168] 两个或两个以上二进制缓冲器20中的每一二进制缓冲器20恰好与一个二进制解码器22连接，且每一二进制解码器仅与一个二进制缓冲器20连接。每一二进制解码器22自单独的部分位串流24读取码字组23，该码字组23表示位的序列。二进制解码器将码字组23转换成二进制21的序列，二进制21的该序列被发送至经连接的二进制缓冲器20。整体解码算法将两个或两个以上部分位串流24转换成若干经解码源符号，其中部分位串流的数目等于二进制缓冲器及二进制解码器的数目，且经由对于新的源符号的请求来触发源符号的解码。在实施例中，二进制解码器22将可变量目的位的码字组23转换成可变数目的二进制21的序列。以上PIPE实施例的一个优点为，可平行(例如，对于机率测度的不同群组而言)执行来自两个或两个以上部分位串流的二进制的解码，该优点减少若干实施的处理时间。

[0169] 以上PIPE解码实施例的另一优点为，可针对参数17的不同集合具体设计经由二进制解码器22执行的二进制解码。特定言的，可针对经估计机率的不同群组最佳化(依据编码效率及/或复杂性)二进制编码及解码。一方面，此举允许降低相对于具有类似编码效率的算术编码算法的编码/解码复杂性。另一方面，此举允许改良相对于具有类似编码/解码复杂性的VLC编码算法的编码效率。在实施例中，二进制解码器22针对当前二进制请求的两个可能二进制值17中的一个二进制值17的机率的估计的测度的不同群组实施不同解码算法(亦即，将二进制序列映射至码字组上)。在另一实施例中，二进制解码器22针对当前所请求的二进制的更小机率或更大机率的二进制值的机率的估计的测度的不同群组实施不同解码算法。在另一实施例中，二进制解码器22针对不同通道保护码实施不同解码算法。在另一实施例中，二进制解码器22针对不同加密方案实施不同解码算法。在另一实施例中，二进制解码器22针对当前所请求的二进制的两个可能二进制值17中的一个二进制值17的机率的估计的测度的通道保护码及群组的不同组合实施不同解码算法。在另一实施例中，二进制解码器22针对当前所请求的二进制的更小机率或更大机率的二进制值17的机率的估计的测度的通道保护码及群组的不同组合实施不同解码算法。在另一实施例中，二进制解码器22针对当前所请求的二进制的两个可能二进制值17中的一个二进制值17的机率的估计的测度的加密方案及群组的不同组合实施不同解码算法。在另一实施例中，二进制解码器22针对当前所请求的二进制的更小机率或更大机率的二进制值17的机率的估计的测度的加密方案及群组的不同组合实施不同解码算法。

[0170] 二进制解码器22在编码器侧执行相应的二进制编码器的逆映像。

[0171] 在实施例中，二进制解码器22或一或更多二进制解码器表示二进制算术解码引擎。

[0172] 在另一实施例中，二进制解码器22或一或更多二进制解码器表示熵解码器，该熵解码器将码字组23直接映像至二进制21的序列上。该映像可有效率地实施且不需要复杂的算术编码引擎。码字组映像至二进制的序列上必须是唯一的。在实施例中，码字组23映射至二进制21的序列上是双射的。在另一实施例中，二进制解码器10或一或更多二进制解码器表示熵解码器，该熵解码器将可变长度码字组23直接映像至二进制21的可变长度序列中。在实施例中，输入码字组表示无冗余度码，诸如，一般的霍夫曼码或典型的霍夫曼码。将无冗余度码双射映射至二进制序列的两个实例图标于表3中。在另一实施例中，输入码字组表示适合于错误侦测及错误恢复的冗余码。在另一实施例中，输入码字组表示加密码。

[0173] 二进制解码器22或一或更多二进制解码器可表示熵解码器，该熵解码器将固定长度码字组23直接映像至二进制21的可变长度序列上。或者二进制解码器22或一或更多二进制解码器表示熵解码器，该熵解码器将可变长度码字组23直接映像至二进制21的固定长度序列上。

[0174] 因此，图3及图4图标用于编码源符号1的序列的PIPE编码器及用于重建源符号1的序列的PIPE解码器的实施例。亦即，图3的PIPE编码器可用作图1A中的PIPE编码器104，其中二进制编码器2充当符号化器122、参数分配器4充当分配器114、二进制缓冲选择器6充当选择器120，且串行连接的二进制缓冲器8及二进制编码器10的对充当熵编码器116中的熵编码器116，该熵编码器116中的每一熵编码器116输出对应于图1A中位串流118的位串流12。如自图3与图1A-图1C的比较而变得清晰的，图1A的分配器114可使该分配器114的输入以替代性方式连接至符号化器122的输入侧，而非该符号化器122的输出侧。类似地，图4的PIPE解码器可用作图2A中的PIPE解码器202，其中部分位串流24对应于图2A和图2B中的位串流216、串行连接的缓冲器20及二进制解码器22的对对应于个别的熵解码器210、二进制缓冲选择器18充当选择器214、参数分配器16充当分配器212且二进制编码器14充当去符号化器222。此外，图2A与图4之间的比较阐明去符号化器222、分配器212与选择器214之间的互连可以不同方式配置，以便根据替代性实施例，修正图2A的连接以对应于图4中所示的连接。

[0175] 图3的PIPE编码器包含分配器4，该分配器4经配置以将若干参数5分配予字母符号3的序列中的每一字母符号。该分配基于字母符号的序列的先前字母符号内含有的信息，诸如，语法元素1的类别，当前字母符号属于该类别的表示，诸如，二进制化，且根据语法元素1的语法结构，该类别为当前预期的，该种预期又是可自先前语法元素1及字母符号3的历史推断的。此外，编码器包含多个熵编码器10及选择器6，该多个熵编码器10中的每一熵编码器10经配置以将转发至相应熵编码器的字母符号3转换成相应位串流12，该选择器6经配置以将每一字母符号3转发至该多个熵编码器10中的选定的熵编码器10，该选择取决于分配予相应字母符号3的参数5的数目。图4的PIPE解码器包含：多个熵解码器22，该多个熵解码器22中的每一熵解码器22经配置以将相应位串流23转换成字母符号21；分配器16，该分配器16经配置以基于字母符号的序列的先前重建的字母符号内含有的信息，将若干参数17分配予待重建的字母符号的序列中的每一字母符号15(参见图4中的26及27)；以及选择器18，该选择器18经配置以自该多个熵解码器22中的选定的熵解码器22检索待重建的字母符号

的序列中的每一字母符号，该选择取决于定义至相应字母符号的参数的数目。分配器16可经配置以使得分配予每一字母符号的参数的数目包含或为相应字母符号可假定的可能字母符号值之中的分布机率的估计的测度。待重建的字母符号的序列可为二进制字母表，且分配器16可经配置以使得机率分布的估计由该二进制字母表的两个可能二进制值的更小机率或更大机率二进制值的机率的估计的测度及指定对于两个可能二进制值中的哪个表示更小机率或更大机率的二进制值的估计的识别符组成。分配器16可进一步经配置以基于待重建的字母符号的序列的先前重建的字母符号内含有的信息，在内部将上下文分配予待重建的字母符号15的序列中的每一字母符号，其中每一上下文具有与该上下文相关联的相应机率分布估计，且分配器16可进一步经配置以基于先前重建的字母符号的符号值，使每一上下文的机率分布估计适应实际符号统计学，相应上下文被分配予该先前重建的字母符号。该上下文可考虑语法元素(诸如)在视讯或图像编码中乃至在财务应用程序情况下的表中所属于的空间关系或邻近位置。随后，可基于与分配予相应字母符号的上下文相关联的机率分布估计，诸如经由对多个机率分布估计代表中的一个机率分布估计代表量化与分配予相应字母符号的上下文相关联的机率分布估计，以获取该机率分布的估计的测度，来决定每一字母符号的机率分布的估计的测度。选择器可经配置以使得在该多个熵编码器与该多个机率分布估计代表之间定义满射关联，亦即，每一熵编码器具有与该熵编码器相关联的至少一个机率分布估计代表，但多于一个机率分布估计代表可与一个熵编码器相关联。根据实施例，该关联甚至可以为双射的。选择器18可经配置以取决于字母符号的序列的先前重建的字母符号，以预定的决定性方式随时间改变自机率分布估计的范围至该多个机率分布估计代表的量化映射。亦即，选择器18可改变量化阶大小，亦即，映射至个别机率标示上的机率分布的区间，该个别机率标示又可以满射方式与个别熵解码器相关联。该多个熵解码器22又可经配置以响应量化映像的改变，调适该多个熵解码器22将字母符号转换成位串流的方式。举例而言，可针对相应机率分布估计量化区间内的某一机率分布估计来最佳化(亦即，可具有最佳压缩率)每一熵解码器22，且每一熵解码器22可在后者改变的后改变该熵解码器22的码字组/符号序列映像，以便调适相应机率分布估计量化区间内的此某一机率分布估计的位置，以进行最佳化。选择器可经配置以改变量化映像，以使得自该多个熵解码器检索字母符号的率变得不太分散。关于二进制编码器14，注意到若语法元素已为二进制，则可远离该二进制编码器14。此外，取决于解码器22的类型，缓冲器20的存在并非必需的。此外，可在解码器内整合缓冲器。

[0176] 到现在为止，以上已关于图3及图4描述图1A及图2A和图2B中的PIPE编码器104及PIPE解码器202的更详细的实施例，若该PIPE编码器104及PIPE解码器202较容易地在图1A及图2A和图2B的设备中实施，则该PIPE编码器104及PIPE解码器202产生平行位串流输出，在该平行位串流输出中平行输送VLC部分位串流及PIPE部分位串流。在下文中，描述如何组合PIPE部分位串流，以随后沿着VLC位串流并行传输，或其中二次交错两种位串流(亦即，VLC位串流及交错的PIPE位串流)的可能性。

[0177] 有限源符号序列的终止

[0178] 在PIPE编码器及解码器的实施例中，针对源符号的有限集合执行编码及解码。通常编码某一数量的数据，诸如，静止影像、视讯序列的讯框或字段、影像的片段、视讯序列的讯框或字段的片段或连续的音讯取样的集合等。对于源符号的有限集合而言，大体上，必须

终止在编码器侧产生的部分位串流，亦即，必须确保可自所传输或储存的部分位串流解码所有的源符号。在最后的二进制插入至相应的二进制缓冲器8中的后，二进制编码器10必须确保完整的码字组被写入至部分位串流12。若二进制编码器10表示二进制算术编码引擎，则必须终止算术码字组。若二进制编码器10表示熵编码器，该熵编码器实施将二进制序列直接映像至码字组上，则在将最后的二进制写入至二进制缓冲器的后储存于二进制缓冲器中的二进制序列可能不表示与码字组相关联的二进制序列(亦即，该二进制序列可能表示与码字组相关联的两个或两个以上二进制序列的前缀)。在此情况下，必须将与二进制序列相关联的码字组中的任一码字组写入至部分位串流，该二进制序列在二进制缓冲器中含有作为前缀的二进制序列(必须清除该二进制缓冲器)。经由将具有特定或任意值的二进制插入至二进制缓冲器中，直至写入码字组为止，可执行此举。在较佳实施例中，二进制编码器选择具有最小长度的码字组中的一个码字组(除相关联二进制序列必须在二进制缓冲器中含有作为前缀的二进制序列的性质之外)。在解码器侧，二进制解码器22可解码比部分位串流中最后的码字组所需要的二进制更多的二进制；并非由二进制缓冲选择器18请求该二进制，且废除并忽略该二进制。经由对于经解码源符号的请求来控制符号的有限集合的解码；若不针对大量数据请求另外的源符号，则终止解码。

[0179] 部分位元串流的传輸及多工

[0180] 可单独传输经由PIPE编码器产生的部分位串流12，或可将该部分位串流12多任务成单一位串流，或可将部分位串流的码字组交错于单一位串流中。

[0181] 在实施例中，将大量数据的每一部分位串流写入至一个数据封包。该大量数据可为源符号的任意集合，诸如，静止图像、视讯序列的字段或讯框、静止图像的片段、视讯序列的字段或讯框的片段或音讯取样的讯框等。

[0182] 在另一较佳实施例中，将大量数据的部分位串流中的两个或两个以上部分位串流或大量数据的所有部分位串流多任务成一个数据封包。含有经多任务部分位串流的数据封包的结构图标于图5中。亦即，图5中所示的数据封包可分别为中间交错串流132及234的部分。

[0183] 数据封包300由标头及每一部分位串流的数据的一个分区组成(对于所考虑量的数据而言)。数据封包的标头300含有将数据封包(的剩余部分)分割成位串流数据302的区段的指示。除对于分割的指示之外，标头可含有额外的信息。在实施例中，对于分割数据封包的指示为在位或字节的单元或多个位或多个字节中数据区段开始的位置。在实施例中，相对于数据封包的开始或相对于标头的末端或相对于先前的数据封包的开始，将数据区段开始的位置编码为数据封包的标头中的绝对值。在另一实施例中，将数据区段开始的位置区别编码，亦即，仅将数据区段的实际开始与对于数据区段开始的预测之间的差异编码。基于已知或已传输的信息，诸如，数据封包的总大小、标头的大小、数据封包中数据区段的数目、居先数据区段开始的位置，可获得该预测。在一实施例中，不将第一数据封包开始的位置编码，而是基于数据封包标头的大小推断第一数据封包开始的该位置。在解码器侧，经传输的分割指示用于获得数据区段的开始。数据区段随后用作部分位串流，且将该数据区段中含有的数据按顺序的次序馈送至相应的二进制解码器中。

[0184] 存在用于将部分位串流12多任务成数据封包的若干替代性实施例。特定言的针对部分位串流的大小十分相似的情况可减少所需要的边信息的一个替代性实施例图示于图6

中。将数据封包(亦即,不具有该数据封包310的标头311的数据封包310)的有效负载以预定义方式分割成区段312。例如,可将数据封包有效负载分割成相同大小的区段。随后每一区段与部分位串流相关联或与部分位串流313的第一部分相关联。若部分位串流大于相关联数据区段,则将该部分位串流的剩余部分314置放至其它数据区段末端处的未使用空间中。可以按逆向次序(自数据区段末端开始)插入位串流的剩余部分的方式执行此举,该方式减少边信息。部分位串流的剩余部分与数据区段的关联及当向数据区段添加多于一个剩余部分时,一或更多剩余部分的起始点必须在位串流内部用讯号发出,例如在数据封包标头中。

[0185] 可变长度码字组的交错

[0186] 对于一些应用而言,一个数据封包中部分位串流12(针对大量源符号)的以上描述的多任务可具有以下缺点:一方面,对于小数据封包而言,发出分割讯号所需要的边信息的位的数目可相对于部分位串流中的实际数据而变得显著,此状况最终降低编码效率。另一方面,多任务可能不适合于需要低迟延的应用(例如,对于视讯会议应用而言)。由于之前并不知道分区开始的位置,故使用所述多任务,PIPE编码器无法在完全产生部分位串流之前开始传输数据封包。此外,一般而言,PIPE解码器必须等待,直至该PIPE解码器在该PIPE解码器可开始解码数据封包之前接收最后的数据区段的开始为止。对于作为视讯会议系统的应用而言,该延迟可合计为若干视讯图像的系统的额外的总延迟(特定言的针对接近传输位率且针对几乎需要两个图像之间的时间区间以编码/解码图像的编码器/解码器),该额外的总延迟对于该应用而言十分关键。为了克服某些应用的缺点,可以将经由两个或两个以上二进制编码器产生的码字组交错至单一位串流中的方式配置PIPE编码器。可将具有经交错码字组的位串流直接发送至解码器(当忽略较小的缓冲延迟时,参见下文)。在PIPE解码器侧,两个或两个以上二进制解码器自位串流按解码次序直接读取码字组;解码可以首先接收的位开始。此外,发出部分位串流的多任务(或交错)讯号不需要任何边信息。

[0187] 具有码字组交错的PIPE编码器的基本结构图标于图7中。二进制编码器10不将码字组直接写入至部分位串流,而是二进制编码器10与单一码字组缓冲器29连接,自该单一码字组缓冲器29,将码字组按编码次序写入至位串流34。二进制编码器10将对于一或更多新的码字组缓冲项28的请求发送至码字组缓冲器29,且二进制编码器10随后将码字组30发送至码字组缓冲器29,该码字组30储存于经保留的缓冲项中。码字组缓冲器29的(一般而言可变长度)码字组31经由码字组写入器32来存取,该码字组写入器32将相应的位33写入至所产生的位串流34。码字组缓冲器29作为先进先出缓冲器操作;较早保留的码字组项被较早写入至位串流。

[0188] 在另一一般化中,多个码字组缓冲器及部分位串流12是可行的,其中码字组缓冲器的数目少于二进制编码器的数目。二进制编码器10在码字组缓冲器29中保留一或更多码字组,藉此一或更多码字组在码字组缓冲器中的保留经由经连接的二进制缓冲器8中的必然事件来触发。在实施例中,以PIPE解码器可瞬时解码位串流34的方式操作码字组缓冲器29,该位串流34分别对应于图1A中的132及图2A和图2B中的134。将码字组写入至位串流的编码次序与将相应的码字组保留于码字组缓冲器中的次序相同。在实施例中,每一二进制编码器10保留一个码字组,其中该保留经由经连接的二进制缓冲器中的必然事件来触发。在另一实施例中,每一二进制编码器10保留多于一个码字组,其中该保留经由经连接的二进制缓冲器中的必然事件来触发。在另外的实施例中,二进制编码器10保留不同量的码字

组，其中经由特定二进制编码器保留的码字组的量可取决于特定二进制编码器及/或特定二进制编码器/二进制缓冲器的其它性质(诸如，相关联机率测度、已写入位的数目等)。

[0189] 在实施例中，如下操作码字组缓冲器。若将新的二进制7发送至特定二进制缓冲器8，且该二进制缓冲器中已储存二进制的数目为零，且当前不存在任何码字组保留于二进制编码器的码字组缓冲器中，该二进制编码器与该特定二进制缓冲器连接，则经连接的二进制编码器10将请求发送至该码字组缓冲器，经由该码字组缓冲器，一或更多码字组项保留于该特定二进制编码器的码字组缓冲器29中。码字组项可具有可变数目的位；通常经由相应的二进制编码器的最大码字组大小给定缓冲项中位的数目的上临界。经由二进制编码器产生的一或更多接下来的码字组(针对该一或更多接下来的码字组已保留一或更多码字组项)储存于码字组缓冲器的经保留的一或更多项中。若特定二进制编码器的码字组缓冲器中的所有经保留缓冲项填充有码字组，且将下一个二进制发送至与该特定二进制编码器连接的二进制缓冲器，则一或更多新的码字组保留于该特定二进制编码器的码字组缓冲器中等。码字组缓冲器29以某一方式表示先进先出缓冲器。缓冲项按顺序的次序保留。较早保留的相应缓冲项所针对的码字组被较早地写入至位串流。码字组写入器32连续地或者在将码字组30写入至码字组缓冲器29的后检查码字组缓冲器29的状态。若第一缓冲项含有完整的码字组(亦即，该缓冲项未经保留，而是该缓冲项包括码字组)，则自码字组缓冲器20移除相应的码字组31及相应的缓冲项，且将码字组的位33写入至位串流。重复此程序，直至第一缓冲项不含有码字组(亦即，该码字组为经保留或自由的)为止。在解码程序的末端，亦即，若已处理考虑量的数据的所有源符号，则必须清除码字组缓冲器。对于彼清除程序而言，针对每一二进制缓冲器/二进制编码器应用下文作为第一步骤：若二进制缓冲器确实含有二进制，则添加具有特定或任意值的二进制，直至所得二进制序列表示与码字组相关联的二进制序列为止(如以上所批注，添加二进制的一个较佳方式为，添加产生与二进制序列相关联的最短可能码字组的该二进制值或该二进制值中的一个二进制值，该二进制序列含有二进制缓冲器的原有内容作为前缀)，随后将该码字组写入至相应的二进制编码器的下一个经保留的缓冲项且清空(相应的)二进制缓冲器。若针对一或更多二进制编码器已保留多于一个缓冲项，则码字组缓冲器可仍含有经保留的码字组项。在彼状况下，该码字组项填充有相应的二进制编码器的任意但有效的码字组。在较佳实施例中，插入最短有效码字组或该最短有效码字组中的一个最短有效码字组(若存在多个)。最终，将码字组缓冲器中所有剩余码字组写入至位串流。

[0190] 码字组缓冲器的状态的两个实例图标于图8A和图8B中。在实例(a)中，码字组缓冲器含有2个项，该2个项填充有码字组及5个经保留项。此外，标记下一个自由缓冲项。第一项填充有码字组(亦即，二进制编码器2刚刚将码字组写入至先前保留的项)。在下一个步骤中，将自码字组缓冲器移除此码字组，且将此码字组写入至位串流。由于仅保留第一缓冲项，但无码字组写入至此项，故随后，二进制编码器3的首先保留的码字组为第一缓冲项，但无法自码字组缓冲器移除此项。在实例(b)中，码字组缓冲器含有3个项，该3个项填充有码字组及4个经保留项。将第一项标记为经保留，因此码字组写入器无法将码字组写入至位串流。尽管在码字组缓冲器中含有3个码字组，但码字组写入器必须等待，直至将码字组写入至二进制编码器3的首先保留的缓冲项为止。注意到必须按保留码字组的次序写入该码字组，以能够使解码器侧的程序逆向(参见下文)。

[0191] 具有码字组交错的PIPE解码器的基本结构图标于图9中。二进制解码器10不直接自单独的部分位串流读取码字组，而是二进制解码器10连接至位缓冲器38，自该位缓冲器38按编码次序读取码字组37。应注意，由于亦可自位串流直接读取码字组，故未必需要位缓冲器38。位缓冲器38主要包括于处理链的明显单独的不同态样的说明中。将具有经交错的码字组的位串流40的位39顺序地插入至位缓冲器38中，该位串流40因此对应于图2A和图2B中的位串流234，该位缓冲器38表示先进先出缓冲器。若特定二进制解码器22接收一或更多二进制序列35的请求，则二进制解码器22自位缓冲器38经由位请求36读取一或更多码字组37。PIPE解码器可瞬时解码源符号。注意到，PIPE编码器(如以上所述)必须经由适当地操作码字组缓冲器来确保按与经由二进制解码器请求码字组的次序相同的次序将该码字组写入至位串流。在PIPE解码器处，整个解码程序经由源符号的请求来触发。作为经由特定二进制编码器保留在编码器侧的码字组的数目与经由相应的二进制解码器读取的码字组的数目的参数必须相同。

[0192] 在另一一般化中，多个码字组缓冲器及部分位串流是可行的，其中位缓冲器的数目少于二进制解码器的数目。二进制解码器22自位缓冲器38一次读取一或更多码字组，藉此自该位缓冲器读取一或更多码字组经由经连接二进制缓冲器20中的必然事件来触发。在实施例中，当将对于二进制19的请求发送至特定二进制缓冲器20且二进制缓冲器不含有任何二进制时，以读取一或更多码字组的方式操作解码器。但例如，若二进制缓冲器中的二进制的数目低于预定义临界，则亦可能经由其它事件来触发读取码字组。在实施例中，每一二进制解码器22读取一个码字组，其中该读取经由经连接的二进制缓冲器中的必然事件来触发。在另一实施例中，每一二进制解码器22读取多于一个码字组，其中该读取经由经连接的二进制缓冲器中的必然事件来触发。在另一实施例中，二进制解码器22读取不同量的码字组，其中经由特定二进制解码器读取的码字组的量可取决于特定二进制解码器及/或特定二进制解码器/二进制缓冲器的其它性质(诸如，相关联机率测度、已读取位的数目等)。

[0193] 在实施例中，如下操作自位缓冲器读取码字组。若将新的二进制请求19自二进制缓冲选择器18发送至特定二进制缓冲器20，且二进制缓冲器中二进制的数目为零，则经连接的二进制解码器22经由对位缓冲器38的位请求36自位缓冲器38读取一或更多码字组37。二进制解码器22将读取的码字组37转换成二进制21的序列，且二进制解码器22将该二进制序列储存于经连接的二进制缓冲器20中。作为对二进制请求19的最终响应，自二进制缓冲器20移除首先插入的二进制，且将首先插入的该二进制发送至二进制缓冲选择器18。作为对进一步二进制请求的响应，移除二进制缓冲器中的剩余二进制，直至清空二进制缓冲器为止。额外的二进制请求触发二进制解码器自位缓冲器读取一或更多新的码字组等。位缓冲器38表示预定义大小的先进先出缓冲器，且位缓冲器38连续地填充有来自位串流40的位39。为了确保以与经由解码程序请求码字组的方式相同的方式将该码字组写入至位串流，可以以上所述方式操作编码器侧的码字组缓冲器。

[0194] 因此，该多个熵解码器中的每一熵解码器可为可变长度解码器，该可变长度解码器经配置以将固定长度的码字组映射至可变长度的符号序列，且可提供诸如码字组缓冲器43的输出的码字组项，以接收经交错码字组的单一串流。该多个熵解码器22可经配置以按顺序的次序自码字组项检索码字组，该顺序的次序取决于经由选择器18自该多个熵解码器检索的待重建的符号序列中的符号产生自相应熵解码器处新的码字组映像的新的符号序

列的次序。

[0195] 具有低延迟限制的可变长度码字组的交错

[0196] PIPE编码的所描述的码字组交错不需要将任何分割信息作为边信息发送。且由于码字组交错于位串流中，故延迟一般较小。然而，并不保证遵照特定延迟限制(例如，由储存于码字组缓冲器中的位的最大数目指定的特定延迟限制)。此外，码字组缓冲器所需要的缓冲区大小可在理论上变得十分大。在考虑第8(b)图中的实例时，有可能没有另外的二进制发送至二进制缓冲器3，因此二进制编码器3将不将任何新的码字组发送至码字组缓冲器，直至应用数据封包末端处的清除程序为止。随后在可将二进制编码器1及2的所有码字组写入至位串流之前，二进制编码器1及2的所有码字组将必须等待，直至数据封包的末端为止。可经由向PIPE编码程序(且亦向随后描述的PIPE解码程序)添加另一机制来防止此缺陷。彼额外的机制的基本概念为，若与延迟有关的测度或延迟的上限(参见下文)超过指定临界，则经由清除相应的二进制缓冲器(使用与数据封包的末端处的机制相似的机制)，来填充首先保留的缓冲项。经由此机制，减少等待的缓冲项的数目，直至相关联延迟测度小于指定临界为止。在解码器侧，必须废除已在编码器侧插入以遵照延迟限制的二进制。对于二进制的此废除而言，可使用基本上与编码器侧的机制相同的机制。在下文中描述此延迟控制的两个实施例。

[0197] 在一个实施例中，延迟的测度(或延迟的上限)为码字组缓冲器中有效缓冲项的数目，其中有效缓冲项的数目为经保留缓冲项的数目加上含有码字组的缓冲项的数目。注意到由于若第一缓冲项含有码字组，则将此码字组写入至位串流，故该第一缓冲项始终为经保留缓冲项或自由缓冲项。若例如，最大允许缓冲延迟(由应用决定的)为D位，且所有二进制编码器的最大码字组大小为L，则在不违反延迟限制的情况下在码字组缓冲器中可含有的码字组的最大数目的下限可经由N＝D/L来计算。系统不需要以位计的延迟测度D，但码字组的最大数目N必须对编码器与解码器两者皆已知。在实施例中，码字组缓冲项的最大数目N由应用固定。在另一实施例中，码字组缓冲项的最大数目N在位串流内部用讯号发出，例如，在数据封包的标头(或片段标头)中或在参数集中，该参数集包括于该位串流中。若二进制编码器10将保留一或更多新的缓冲项的请求发送至码字组缓冲器29，则在保留新的码字组缓冲项之前执行以下程序(亦即，若经由一个请求保留多个码字组缓冲项，则多次执行该程序)：若当前有效缓冲项的数目加上1(考虑接下来将保留的缓冲项)大于码字组缓冲项的最大数目N，则经由下文中描述的程序清除第一缓冲项(该第一缓冲项为保留的)，直至当前有效缓冲项的数目加上1小于或等于码字组缓冲项的最大数目N为止。清除经保留缓冲项类似于数据封包的末端处的清除：经由向经连接的二进制缓冲器8添加具有特定或任意值的二进制，直至所得二进制序列表示与码字组相关联的二进制序列为止，来清除已保留的相应第一缓冲项的二进制编码器10，随后将该码字组写入至经保留缓冲项，且最终向位串流添加该码字组(同时清空二进制缓冲器且移除先前保留的缓冲项)。如以上所提及，向二进制缓冲器添加二进制的一个较佳方式为，添加产生最短可能码字组的彼等二进制。在解码器侧，执行用于废除为遵照延迟限制而添加的二进制的相似程序。因此，解码器维持计数器C，该计数器C给自位缓冲器读取的码字组计数(可在位缓冲器中维持此计数器)。在解码数据封包开始时初始化(例如，具有零)此计数器C，且在读取码字组的后此计数器C增加一。此外，每一二进制解码器22含有计数器Cx，该计数器Cx在相应的二进制解码器22读取最后的

码字组之前储存码字组计数器C的值。亦即，当特定二进制解码器22读取新的码字组时，将该特定二进制解码器22的计数器Cx设定成等于C作为第一步骤，且随后自位缓冲器读取码字组。当将二进制请求19发送至特定二进制缓冲器20，且整体码字组计数器C与经连接的二进制解码器22的计数器Cx之间的差值(C-Cx)大于码字组缓冲项的最大数目N时，废除且忽略当前储存于特定二进制缓冲器20中的所有二进制。除彼额外的步骤之外，如以上所述操作解码。若清空二进制缓冲器20(因为已移除所有二进制，或者因为在接收二进制请求的后，低延迟机制在第一步骤中废除所有二进制)，二进制请求19被发送至该二进制缓冲器20，则经连接的二进制解码器22自位缓冲器38等读取一或更多新的码字组。

[0198] 在另一实施例中，延迟的测度(或延迟的上限)为码字组缓冲器中有效缓冲项的最大码字组长度的和，其中特定缓冲项的最大码字组长度取决于与彼缓冲项相关联的经解码二进制。作为说明，在实例中用6指示缓冲项的最大码字组长度。又注意到由于若第一缓冲项含有码字组，则将此码字组写入至位串流，故该第一缓冲项始终为经保留缓冲项或自由缓冲项。使最大经允许缓冲延迟(由应用决定的)为D位。此最大缓冲延迟D必须对编码器与解码器两者皆已知。在较佳实施例中，由应用固定最大缓冲延迟D。在另一较佳实施例中，最大缓冲延迟D在位串流内部用讯号发出，例如，在数据封包的标头(或片段标头)中或在参数集中，该参数集包括于该位串流中。可在位或字节的单元或多个位或多个字节中用讯号发出该最大缓冲延迟D。若二进制编码器10将保留一或更多新的缓冲项的请求发送至码字组缓冲器29，则在保留新的码字组缓冲项之前执行以下程序(亦即，若经由一个请求保留多个码字组缓冲项，则多次执行该程序)。

[0199] 若所有当前有效缓冲项的最大码字组长度的和加上将被保留的缓冲项的最大码字组长度大于最大缓冲延迟D，则经由以上所述程序清除该第一缓冲项(该第一缓冲项为经保留的)，直至所有有效缓冲项的最大码字组长度的和加上将被保留的缓冲项的最大码字组长度小于或等于最大缓冲延迟D为止。例如，考虑第8(b)图中的实例。所有当前有效缓冲项的最大码字组长度的和为29。假定将最大缓冲延迟D设定成等于32。由于29+3并不大于32，故若经由二进制编码器2保留的下一个缓冲项的最大码字组长度等于3，则未清除第一缓冲项。但由于29+7大于32，故若经由二进制编码器1保留的下一个缓冲项的最大码字组长度等于7，则第一缓冲项被清除。如以上所述执行经保留缓冲项的清除(经由向相应的二进制缓冲器添加具有特定或任意值的二进制)。

[0200] 在解码器侧，执行用于废除为遵照延迟限制而添加的二进制的相似程序。因此，解码器维持计数器C，该计数器C给自位缓冲器读取的码字组的最大码字组长度计数(可在位缓冲器中维持此计数器)。注意到，与不同二进制解码器相关联的最大码字组长度可不同。在解码数据封包开始时初始化(例如，具有零)该计数器C，且在读取码字组的后该计数器C增加。此计数器未增加读取的码字组的实际长度，而是增加该读取的码字组的最大长度。亦即，若经由特定二进制解码器读取码字组，且与该特定二进制解码器使用的码字组表相关联的最大码字组长度为Lx(不同的二进制解码器可与不同的最大码字组长度相关联)，则计数器C增加Lx。除整体计数器C之外，每一二进制解码器22含有计数器Cx，该计数器Cx在相应的二进制解码器22读取最后的码字组之前储存码字组计数器C的值。亦即，当特定二进制解码器22读取新的码字组时，将该特定二进制解码器22的计数器Cx设定成等于C作为第一步骤，且随后自位缓冲器读取码字组。当将二进制请求19发送至特定二进制缓冲器20，且整体

码字组计数器C与经连接的二进制解码器22的计数器Cx之间的差值(C-Cx)大于最大缓冲延迟D时，废除且忽略当前储存于特定二进制缓冲器20中的所有二进制。除彼额外的步骤之外，如以上所述操作解码。若清空二进制缓冲器20(因为已移除所有二进制，或者因为在接收二进制请求的后，低延迟机制在第一步骤中废除所有二进制)，二进制请求19被发送至该二进制缓冲器20，则经连接的二进制解码器22自位缓冲器38等读取一或更多新的码字组。

[0201] 因此，该多个熵解码器22及选择器18可经配置以间歇废除符号序列的后缀，以便不参与形成待重建的符号的序列29。可在事件处执行间歇废除，在该事件处，经由相应熵解码器自码字组项的两个连续的码字组检索之间的多个熵解码器自码字组项检索的若干码字组满足预定标准。该多个熵编码器及码字组缓冲器可又经配置以经由具有当前转发但尚未映像的符号作为前缀的全都符合(don't-care)的符号将当前转发但尚未映像的符号间歇地延伸至有效的符号序列；该多个熵编码器及码字组缓冲器可又经配置以将如此延伸的符号序列映像至码字组中，该多个熵编码器及码字组缓冲器可又经配置以将因此获取的码字组输入至经保留码字组项中，且该多个熵编码器及码字组缓冲器可又经配置以清除该码字组项。该间歇延伸、输入及清除可发生在若干经保留码字组项加上具有在该处输入的码字组的若干码字组项满足一预定标准的若干事件处。此预定标准可考虑该多个编码器/解码器对的码字组最大长度。

[0202] 对于一些架构而言，码字组交错的以上所述较佳实施例可能产生关于解码复杂性的缺陷。如图9中所示，所有二进制解码器22自单一位缓冲器38读取码字组(在一般情况下，可变长度码字组)。由于必须按正确次序读取码字组，故无法平行执行码字组的读取。彼状况意谓特定二进制解码器必须等待，直至其它二进制解码器完成码字组的读取为止。且当读取可变长度码字组的复杂性相对于(部分平行化)解码程序的剩余部分较显著时，可变长度码字组的此存取可能为整个解码程序的瓶颈。存在所述实施例的一些变化，该变化可用于降低自单一位缓冲存取的复杂性，将在下文中描述该变化中的数个变化。在一个较佳实施例中，存在码字组的单一集合(表示例如无冗余度前缀码)，且用于每一二进制解码器22的码字组的集合为该单一码字组集合的子集。注意到，不同二进制解码器22可使用该单一码字组集合的不同子集。即使二进制解码器22中的一些二进制解码器22所使用的码字组集合相同，该码字组集合与二进制序列的关联因不同的二进制解码器22而不同。在特定实施例中，码字组的相同集合用于所有二进制解码器22。若我们具有单一码字组集合，该单一码字组集合包括所有二进制解码器的码字组集合作为子集，则可在二进制解码器外部执行码字组的剖析，此举可降低码字组存取的复杂性。PIPE编码程序相对于以上所述程序并未改变。经修改的PIPE解码程序图标于图10中。用来自位串流40的位46馈送给单一码字组读取器，且该单一码字组读取器剖析(一般而言可变长度)码字组。将所读取的码字组44插入至码字组缓冲器43中，该码字组缓冲器43表示先进先出缓冲器。二进制解码器22将一或更多码字组41的请求发送至码字组缓冲器43，且作为对此请求的响应，自码字组缓冲器移除一或更多码字组(按顺序的次序)且将该一或更多码字组发送至相应的二进制解码器22。注意到，使用此实施例，可在后台程序中执行可能复杂的码字组剖析，且该剖析不需要等待二进制解码器。二进制解码器存取已剖析码字组，可能复杂的码字组剖析不再为对整体缓冲器的请求的部分。实情为，将已剖析码字组发送至二进制解码器，亦可以将仅码字组标示发送至二进制解码器的方式实施此举。

[0203] 固定长度位元序列的交错

[0204] 可在不自全局位缓冲器38读取可变长度码字组时实现降低PIPE解码器复杂性的另一方式，但实情为，该二进制解码器22始终自全局位缓冲器38读取位的固定长度序列且向局部位缓冲器添加位的该固定长度序列，其中每一二进制解码器22与单独的局部位缓冲器连接。随后自局部位缓冲器读取可变长度码字组。因此，可平行执行可变长度码字组的剖析，仅位的固定长度序列的存取必须以同步方式执行，但位的固定长度序列的此存取通常十分迅速，以便可针对一些架构降低整体解码复杂性。取决于作为二进制解码器、二进制缓冲器或位缓冲器中的事件的某些参数，发送至特定局部位缓冲器的二进制的固定数目可因不同局部位缓冲器而不同，且该固定数目亦可随时间变化。然而，经由特定存取读取的位的数目并不取决于在特定存取期间读取的实际位，此状况为与读取可变长度码字组的重要区别。读取位的固定长度序列经由二进制缓冲器、二进制解码器或局部位缓冲器中的必然事件来触发。例如，当存在于经连接位缓冲器中的位的数目降到预定义临界以下时，有可能请求读取位的新的固定长度序列，其中不同临界值可用于不同位缓冲器。在编码器处，必须保证按相同次序将二进制的固定长度序列插入至位串流中，按该次序在解码器侧自位串流读取二进制的该固定长度序列。亦可能将固定长度序列的此交错与类似于以上阐释的低延迟控制的低延迟控制组合。在下文中，描述位的固定长度序列的交错的较佳实施例。

[0205] 图11图示针对两个或两个以上二进制编码器交错位的固定长度序列的实施例的PIPE编码器结构的说明。与图7中所示实施例相反，二进制编码器10并未与单一码字组缓冲器连接。实情为，每一二进制编码器10与单独的位缓冲器48连接，该单独的位缓冲器48储存相应的部分位串流的位。所有位缓冲器48连接至全局位缓冲器51。全局位缓冲器51连接至位写入器53，该位写入器53自该全局位缓冲器按编码/解码次序移除位52且将所移除的位54写入至位串流55。在特定位缓冲器48或经连接二进制编码器10或二进制缓冲器8中的必然事件的后，位缓冲器48将请求49发送至全局位缓冲器51，经由该请求49，将某一数目的位保留于全局位缓冲器51中。按顺序的次序处理保留固定长度位序列的请求49。全局位缓冲器51以某一方式表示先进先出缓冲器；较早被保留的位被较早地写入至位串流。应注意，不同位缓冲器48可保留不同量的位，该量亦可基于已编码符号而随时间变化；但在将请求发送至全局位缓冲器时已知经由特定请求保留的位的数目。

[0206] 特定言的，如下文中所述操作位缓冲器48及全局位缓冲器51。经由特定位缓冲器48保留的位的量表示为Nx。位的此数目Nx可因不同位缓冲器48而不同，且该数目亦可随时间变化。在较佳实施例中，经由特定位缓冲器48保留的位的数目Nx随时间固定。基于位缓冲器48中位的数目Mx、保留请求的位的数目Nx及相关联的最大码字组长度Lx，来触发保留固定数目Nx的位49。注意到，每一二进制编码器10可与不同最大码字组长度Lx相关联。若将二进制7发送至特定二进制缓冲器8，且该特定二进制缓冲器8是清空的，且Nx个位的不多于一个序列保留于位缓冲器48的全局位缓冲器中，该位缓冲器48与该特定二进制缓冲器连接(经由二进制编码器)，且位的数目Nx与位的数目Mx之间的差值Nx-Mx小于与相应的二进制编码器10相关联的最大码字组长度Lx，则经连接位缓冲器49将保留Nx个位的请求49发送至全局位缓冲器51，位的该数目Nx经由位缓冲器48的保留请求来保留，该位缓冲器48与该特定二进制缓冲器8(经由二进制编码器)连接，位的该数目Mx当前存在于此位缓冲器48中。全局位缓冲器51保留特定位缓冲器48的Nx个位，且全局位缓冲器51为下一次保留增加该全局

位缓冲器51的指针。在Nx个位保留于全局位缓冲器中的后，将二进制7储存于二进制缓冲器8中。若此单一二进制确实已表示与码字组相关联的二进制序列，则二进制编码器10自二进制缓冲器8移除此二进制且将相应的码字组47写入至经连接位缓冲器48。否则(此单一二进制确实已表示与码字组相关联的二进制序列)，经由特定二进制缓冲器8接受另外的二进制7，直至该二进制缓冲器8含有与码字组相关联的二进制序列。在此情况下，经连接的二进制编码器10自二进制缓冲器8移除二进制序列9且将相应的码字组47写入至经连接位缓冲器48。若位缓冲器48中位的所得数目Mx大于或等于经保留位的数目Nx，则将首先写入至位缓冲器48的Nx个位插入至全局位缓冲器51中的先前保留空间中。对于被发送至特定二进制缓冲器8的下一个二进制7而言，执行与以上指定的程序相同的程序；亦即，首先检查新数目的Nx个位是否必须保留于全局位缓冲器中(若Nx-Mx小于Lx)，且随后将该二进制插入至二进制缓冲器8中等。位写入器将全局位缓冲器的固定长度位序列按保留该固定长度位序列的次序写入。若全局位缓冲器51中的第一固定长度项含有实际上已在全局位缓冲器中插入的固定长度位序列(亦即，该第一固定长度项不仅是保留的)，则位写入器53自全局位缓冲器51移除此位序列52的位且将位54写入至位串流。重复此程序，直至全局位缓冲器中的第一固定长度项表示经保留或自由项为止。若全局位缓冲器中的第一固定长度项表示经保留项，则在位写入器53将另外的位54写入至位串流55之前，该位写入器53等待，直至此项填充有实际位为止。

[0207] 在数据封包的末端，如以上所述清除二进制缓冲器。此外，必须经由添加具有特定或任意值的位，直至全局位缓冲器中所有经保留缓冲项被填充且写入至位串流为止，来清除位缓冲器。

[0208] 在图12A和图12B中，图标全局位缓冲器51的可能状态的两个实例。在实例(a)中，图标不同位缓冲器/二进制编码器保留不同数目的位的情况。全局位缓冲器含有具有实际经写入固定长度位序列的3个项及具有经保留固定长度位序列的4个项。第一固定长度项已含有实际位(该实际位一定刚刚经由位缓冲器/二进制编码器2插入)；可移除此项(亦即，相应的8个位)，且将此项写入至位串流。下一个项保留二进制编码器3的10个位，但尚未插入实际位。无法将此项写入至位串流；必须等待此项，直至插入实际位为止。在第二实例(b)中，所有位缓冲器/二进制编码器保留相同数目的位(8个位)。全局位缓冲器含有8个位序列的4个保留及实际经写入8个位序列的3个保留。第一项含有二进制编码器3的8个位的保留。在可将任何新的位写入至位串流之前，位写入器必须等待，直至位缓冲器/二进制编码器3将8个位的实际值写入至此经保留项中。

[0209] 图13图标交错位的固定长度序列的本发明的实施例的PIPE解码器结构的说明。与图9中所示实施例相反，二进制解码器22并未与单一位缓冲器连接。实情为，每一二进制解码器22与单独的位缓冲器58连接，该单独的位缓冲器58储存来自相应的部分位串流的位。所有位缓冲器58连接至全局位缓冲器61。将来自位串流63的位62插入至全局位缓冲器61中。在特定位缓冲器58或经连接二进制解码器22或二进制缓冲器20中的必然事件的后，位缓冲器58将请求59发送至全局位缓冲器61，经由该请求59，自全局位缓冲器61移除位60的固定长度序列且将位60的固定长度序列插入至特定位缓冲器58中。按顺序的次序处理固定长度位序列的请求59。全局位缓冲器61表示先进先出缓冲器；较早地移除较早被插入至全局位缓冲器中的位。应注意，不同位缓冲器58可请求不同量的位，该量亦可基于已解码符号

而随时间变化;但在将请求发送至全局位缓冲器时已知经由特定请求所请求的位的数目。应注意,由于亦可自位串流直接读取码字组,故未必需要全局位缓冲器61。全局位缓冲器61主要包括于处理链的明显单独的不同态样的说明中。

[0210] 在实施例中,如下文中所述操作位缓冲器58及全局位缓冲器61。经由特定位缓冲器58请求且读取的位的量表示为Nx,该量等于经由编码器侧的相应位缓冲器写入至全局位缓冲器的位的量。位的此数目Nx可因不同位缓冲器58而不同,且该数目亦可随时间变化。在本发明的较佳实施例中,经由特定位缓冲器58请求且读取的位的数目Nx随时间固定。基于位缓冲器58中位的数目Mx及相关联的最大码字组长度Lx,触发固定数目Nx的位60的读取。注意到,每一二进制解码器22可与不同最大码字组长度Lx相关联。若将二进制请求19发送至特定二进制缓冲器20,且该特定二进制缓冲器20为清空的,且位缓冲器58中位的数目Mx小于与相应二进制解码器22相关联的最大码字组长度Lx,则经连接位缓冲器58将Nx个位的新的序列的请求59发送至全局位缓冲器61,该位缓冲器58与该特定二进制缓冲器20连接(经由二进制解码器)。作为对此请求的响应,自全局位缓冲器61移除最初Nx个位,且将Nx个位60的此序列发送至位缓冲器58,自该位缓冲器58发送该请求。最终,向相应的位缓冲器58添加Nx个位的此序列。随后自此位缓冲器读取下一个码字组57,且经连接二进制解码器22将相关联二进制序列21插入至经连接二进制缓冲器20中。作为对二进制原有请求19的最终响应,自二进制缓冲器20移除第一二进制,且将此经解码二进制25发送至二进制缓冲选择器18。当将下一个二进制请求19发送至特定二进制缓冲器20且二进制缓冲器不是清空的时,自该二进制缓冲器20移除下一个位。若二进制缓冲器为清空的,但经连接位缓冲器58中位的数目Mx大于或等于相关联最大码字组长度Lx,则自位缓冲器读取下一个码字组,且将新的二进制序列插入二进制缓冲器中,自该二进制缓冲器移除第一位,且将该第一位发送至二进制缓冲选择器。若二进制缓冲器为清空的,且经连接位缓冲器58中位的数目Mx小于相关联最大码字组长度Lx,则自全局位缓冲器61读取Nx个位的下一个序列,且将Nx个位的该下一个序列插入至经连接局部位缓冲器58中,自位缓冲器读取下一个码字组,将新的二进制序列插入二进制缓冲器中,且移除该序列的第一二进制且将序列的该第一二进制发送至二进制缓冲选择器。重复此程序,直至解码所有源符号为止。

[0211] 在数据封包的末端,可能将比解码经请求源符号所需要的二进制及/或位更多的二进制及/或位插入至二进制缓冲器及/或位缓冲器中。废除且忽略二进制缓冲器中的剩余二进制及位缓冲器中的剩余位。

[0212] 具有低延迟限制的固定长度位元序列的交错

[0213] 具有固定长度位序列的交错的PIPE编码器及解码器的所述实施例亦可与以上所述的控制编码器缓冲延迟的方案组合。PIPE编码概念与具有以上所述延迟控制的实施例中的PIPE编码概念相同。若与延迟有关的测度或延迟的上限(参见下文)超过指定临界,则经由清除相应的二进制缓冲器(使用与数据封包的末端处的机制相似的机制),且可能写入用于填充经保留固定长度缓冲项的所有位的额外的位,来填充首先保留的缓冲项。经由此机制,减少等待的缓冲项的数目,直至相关联延迟测度小于指定临界为止。在解码器侧,必须废除已在编码器侧插入以遵照延迟限制的二进制及位。对于二进制及位的此废除而言,可使用基本上与编码器侧的机制相同的机制。

[0214] 在实施例中,延迟的测度(或延迟的上限)为全局位缓冲器中的有效缓冲项中位的

数目，其中有效缓冲项的数目为经保留固定长度缓冲项的数目加上含有已写入位的固定长度缓冲项的数目。注意到由于若第一缓冲项含有经写入位，则将该位写入至位串流，故该第一缓冲项始终为经保留固定长度缓冲项或自由缓冲项。使最大经允许缓冲延迟(由应用决定的)为D位。此最大缓冲延迟D必须对编码器与解码器两者皆已知。在本发明的较佳实施例中，由应用固定最大缓冲延迟D。在本发明的另一较佳实施例中，最大缓冲延迟D在位串流内部用讯号发出，例如，在数据封包的标头(或片段标头)中或在参数集中，该参数集包括于该位串流中。可在位或字节的单元或多个位或多个字节中用讯号发出该最大缓冲延迟D。若二进制编码器10将保留新的固定长度位序列的请求发送至全局位缓冲器51，则在保留新的固定长度缓冲项之前执行以下程序。

[0215] 若全局位缓冲器中的有效缓冲项中的位的数目加上将经由当前保留请求保留的位的数目大于最大缓冲延迟D，则经由下文中所述程序清除第一缓冲项(该第一缓冲项为经保留的)，直至全局位缓冲器中的有效缓冲项中的位的数目加上将经由当前保留请求保留的位的数目小于或等于最大缓冲延迟D为止。清除经保留固定长度缓冲项类似于数据封包的末端处的清除：经由向经连接的二进制缓冲器8添加具有特定或任意值的二进制，直至所得二进制序列表示与码字组相关联的二进制序列为止，来清除与已保留的相应第一缓冲项的位缓冲器48连接的二进制编码器10，随后将该码字组插入至相应的位缓冲器48中。如以上所提及，向二进制缓冲器添加二进制的一个较佳方式为，添加产生最短可能码字组的彼等二进制。若在将码字组写入至经连接位缓冲器且可能将固定长度位序列插入至全局位缓冲器中的后，在位缓冲器中仍存在位(亦即，经写入码字组未完全填充位的经保留固定长度序列)，则向位缓冲器添加具有特定或任意值的另外的位，直至自该位缓冲器移除所有位且将所有位写入至经保留缓冲项为止。最终，在此程序的末端处，自全局位缓冲器移除完整的缓冲项(全局位缓冲器中的第一固定长度项)且将该完整的缓冲项写入至位串流。

[0216] 在解码器侧，执行用于废除为遵照延迟限制而添加的二进制及位的相似程序。因此，解码器维持计数器C，该计数器C给自全局位缓冲器读取的位计数(可在全局位缓冲器中维持此计数器)。在解码数据封包开始时初始化(例如，具有零)该计数器C，且在读取固定长度序列的后该计数器C增加。若自全局位缓冲器61读取Nx个位的固定长度序列，则计数器C增加Nx。除整体计数器C之外，每一位缓冲器58含有计数器Cx，该计数器Cx在将最后的固定长度位序列读取至相应的位缓冲器58中之前储存位计数器C的值。当特定位缓冲器58读取新的固定长度位序列时，将该特定位缓冲器58的计数器Cx设定成等于C作为第一步骤，且随后自全局位缓冲器61读取该固定长度位序列。当将二进制请求19发送至特定二进制缓冲器20，且整体计数器C与经连接的位缓冲器58的计数器Cx之间的差值(C-Cx)大于最大缓冲延迟D时，废除且忽略当前储存于特定二进制缓冲器20中的所有二进制及储存于经连接位缓冲器58中的所有位。除彼额外的步骤之外，如以上所述操作解码。若清空二进制缓冲器20(因为已移除所有二进制，或者因为在接收二进制请求的后，低延迟机制在第一步骤中废除所有二进制)，二进制请求19被发送至该二进制缓冲器20，则经连接的二进制解码器22试图自经连接位缓冲器58读取新的码字组。若位缓冲器58中位的数目小于最大码字组长度，则在读取码字组之前，自全局位缓冲器61读取新的固定长度位序列等。

[0217] 刚刚已关于图7至图13描述与在一方面PIPE编码器104与另一方面PIPE解码器202之间实现交错的位串流路径的可能性有关的实施例。如以上关于图1A-图1C及图2A和图2B

所述，熵编码及解码设备可经由两个分离信道彼此连接，该两个分离信道中的一个分离信道输送VLC位串流112，且该两个分离信道中的另一个分离信道输送经交错的PIPE编码位串流。然而，亦存在交错甚至VLC位串流112以及经PIPE编码位串流118两者的可能性，且稍后将关于图20至图24描述该可能性。然而，在彼描述之前，提供关于PIPE编码方案以及有关如何最佳地再划分机率区间的细节的数学背景，其中将所得个别部分区间分别分配予个别熵编码器116及熵解码器210。

[0218] 如上所述，在PIPE编码时，将离散符号的输入序列的事件空间映射至二进制机率区间小集合上。当使用机率区间的熵编码保持固定且自模型化阶段去耦该熵编码时，源符号的机率模型可为固定或适应性的。可使用十分简单的熵码来编码机率区间中的每一机率区间，该熵码具有霍夫曼码的复杂性位准。机率区间分割熵(PIPE)码的超出率类似于纯算术编码的超出率。

[0219] 熵编码一般可视为无损数据压缩的最一般的形式。无损压缩目的在于用比原始数据表示所需要的位更少的位显示离散数据，但无任何信息损失。可以本文、图形、影像、视讯、音讯、语音、传真、医疗数据、气象数据、财务数据或数字数据的任何其它形式的形式给出离散数据。在许多编码应用中，将原始源数据首先映像至所谓的编码符号上，且随后将该编码符号熵编码。映像至编码符号上可包括量化，在此情况下，整体编码方案为有损的。编码符号s可取M阶(M≥2)字母表$A=\{a_0,\ldots,a_{M-1}\}$的任何值。出于编码符号s的目的，字母表与经估计的机率质量函数(pmf)$\{p_s(a_0),\ldots,p_s(a_{M-1})\}$相关联，且忽略在此pmf中不考虑的编码符号之间的所有相关性。对于该抽象设定而言，熵

[0220] $$H_s=-\sum_{i=0}^{M-1}p_s(a_i)\log_2 p_s(a_i) \qquad (B1)$$

[0221] 为每符号位中预期码字组长度的最大下限，对于编码符号s而言，使用熵编码技术可实现彼状况。几十年来，霍夫曼编码及算术编码已主导实用性熵编码。霍夫曼编码及算术编码为能够接近熵极限(在某种意义上)的实用码的熟知实例。

[0222] 对于固定机率分布而言，霍夫曼码相对容易建构。霍夫曼码的最吸引人的性质为，该霍夫曼码的实施可有效率地经由使用可变长度码(VLC)表来实现。然而，当处理时变源统计学(亦即，变化的符号机率)时，极其需要调适霍夫曼码及该霍夫曼码的相应的VLC表，霍夫曼码及该霍夫曼码的相应的VLC表两者皆依据算法复杂性以及依据实施成本。亦，在具有$p_s(a_i)>0.5$的主要字母值的情况下，相应的霍夫曼码的冗余度(在不使用诸如游程编码的任何字母延伸的情况下)可为相当实质的。霍夫曼码的另一缺陷经由在处理高阶机率模型化情况下可能需要多组VLC表的事实给出。

[0223] 另一方面，尽管算术编码实质上比VLC更复杂，但当应对适应性且高阶机率模型化以及应对高度偏斜机率分布的情况时，该算术编码提供更一致且更充分处理的优点。实际上，此特性基本上由算术编码提供将机率估计的任何给定值以差不多直接的方式映像至所得码字组的部分的至少概念性机制的事实产生。在具备此接口的情况下，算术编码允许一方面机率模型化及机率估计的任务与另一方面实际的熵编码亦即将符号映像至码字组之间的清晰分离。

[0224] 不同于刚刚论述的习知熵编码方案，PIPE编码使用机率区间分割，以下更详细地描述该机率区间分割的数学背景。

[0225] 考虑编码符号的序列{$s_0,\ldots,s_{N-1}$}。自字母表$s_i \in A_i$抽出每一符号。字母表$A_i=\{a_0^i,a_1^i,\ldots\}$含有两个或两个以上字母，该两个或两个以上字母各自与机率估计$p_s(a_m^i)$相关联。机率估计$p_s(a_m^i)$对编码器及解码器已知，且机率估计$p_s(a_m^i)$可为固定或可变的。假定在编码器及解码器处同时估计可变机率。字母表A_i可对于符号的序列而言为相同的，或者不同符号类型与不同字母表相关联。在后者情况下，假定解码器已知序列中每一符号的字母表。此假定被证明是正确的，因为实用源编解码器描述含有语法，该语法规定符号的次序及该符号的字母表。

[0226] 符号的序列{$s_0,\ldots,s_{N-1}$}转换成二进制符号的序列，该二进制符号亦称为二进制。对于每一符号s_i而言，二进制化

[0227] $$b^i=\{b_0^i\ldots\}=\gamma_b^i(s_i) \quad (B2)$$

[0228] 表示将字母表字母a_m^i双射映射至二进制b^i_m的有次序的集合上。该二进制化映射γ_b^i可因不同符号s_i或符号类别而不同。特定符号s_i的每一二进制序列i由一或更多二进制b_k^i组成。在解码器侧，符号s_i可在给定二进制b^i的序列的情况下，经由逆映射$s_i=(\gamma_b^i)^{-1}(\mathbf{b}^i)$来重建。作为二进制化的结果，获取二进制的序列{$b_0,\ldots,b_{B-1}$}二进制的该序列表示源符号的序列{$s_0,\ldots,s_{N-1}$}。

[0229] 所有二进制b_j与相同二进制字母表B＝{0,1}相关联，但$p_1^j=p_0^j$情况下的相应二进制$\mathrm{pmf}\{p_0^j,p_1^j\}$通常不同。二进制$\mathrm{pmf}\{p_0^j,p_1^j\}$可经由更小机率二进制(less probable bin;LPB)值b^j_{LPB}及该LPB值的机率p^j_{LPB}(其中$p^j_{LPB}\le 0.5$)来描述。此二进制机率描述$\{b^j_{LPB},p^j_{LPB}\}$可在给定二进制化映射γ_b^i的情况下自符号字母表的机率估计$p_s(a_m^i)$直接获得。亦可能(且通常较佳地)同时在编码器及解码器侧直接估计$\{b^j_{LPB},p^j_{LPB}\}$。因此，二进制可与基于语法及先前经编码符号或二进制的机率模型(该机率模型亦称为上下文)相关联。且对于每一机率模型而言，可基于用机率模型编码的二进制的值，来估计机率描述$\{b^j_{LPB},p^j_{LPB}\}$。关于H.264的CABAC描述此二进制机率模型化的实例。

[0230] 由于二进制熵函数

[0231] $$H(p)=-p\log_2(p)-(1-p)\log_2(1-p) \quad (B3)$$

[0232] 在p＝0.5周围是对称的，故相同二进制编码器可用于编码与相同LPB机率p^j_{LPB}相关联的所有二进制，而与b^j_{LPB}的值无关。因此，二进制的序列{$b_0,\ldots,b_{B-1}$}转换成编码二进制的序列$\{b_0^c,\ldots,b_{B-1}^c\}$。对于每一二进制$b_j$而言，相应的双射映射$\gamma_c^j$由

[0233] $$b_j^c=\gamma_c^j(b_j)=b_j\oplus b^j_{LPB} \quad (B4)$$

[0234] 指定，其中⊕表示互斥或运算子。在解码器侧，二进制b_j可在给定编码二进制b_j^c及相应的LPB值b^j_{LPB}的情况下经由逆映射$b_j=(\gamma_c^j)^{-1}(b_j^c)=b_j^c\oplus b^j_{LPB}$重建。编码二进制$b_j^c=0$

指定相应的二进制b_j的值等于LPB值b^j_{LPB}，且编码二进制$b^c_j=1$指定相应的二进制的值b_j等于更大机率二进制(more probable bin;MPB)值$1-b^j_{LPB}$。

[0235] 编码二进制的序列$\{b^c_0,\ldots,b^c_{B-1}\}$唯一地表示源符号的序列$\{s_0,\ldots,s_{N-1}\}$，且可用于熵编码的相应机率估计完全经由LPB机率$p^j_{LPB}$(其中$p^j_{LPB}\le 0.5$)来描述。因此，仅半开区间(0,0.5]中的机率需要考虑用于设计编码二进制b^c_i的二进制熵编码器。

[0236] 对于实际的二进制熵编码而言，将编码二进制的序列$\{b^c_0,\ldots,b^c_{B-1}\}$投射至少量机率区间$I_k$上。将LPB机率区间(0,0.5]分割成K个区间$I_k=(p_k,p_{k+1}]$。

[0237] $\bigcup_{k=0}^{K-1} I_k=(0,0.5]$ 且 $I_k\cap I_j=\varnothing$ 对于k≠j (B5)

[0238] K个区间的集合的特征在于K-1个区间界限pk，其中k=1,...,K-1。在不失一般性的情况下，我们针对k=0,...,K假定$p_k<p_{k+1}$。外部区间界限由$p_0=0$及$p_K=0.5$固定且给定。针对每一区间I_k设计简单的非适应性二进制熵编码器。具有相关联LPB机率$p^j_{LPB}\in I_k$的所有编码二进制b^c_j被分配予区间I_k且用相应的经固定熵编码器编码。

[0239] 在以下描述中，所有二进制表示编码二进制b^c_j且所有机率p为LPB机率p^j_{LPB}。

[0240] 为研究机率区间离散化对编码效率的影响，我们假定我们可针对达到熵界限的固定机率设计最佳的熵编码器。每一机率区间$I_k=(p_k,p_{k+1}]$与代表性的机率$p_{I_k}\in I_k$相关联，且相应最佳的熵编码器应达到此代表性的机率的熵极限。在该假定的下，使用区间代表p_{I_k}的最佳的熵编码器编码具有机率p的二进制的率由

[0241]
$$R(p,p_{I_k})=-p\log_2(p_{I_k})-(1-p)\log_2(1-p_{I_k})$$
$$=H(p_{I_k})+(p-p_{I_k})H'(p_{I_k}) \quad (B6)$$

[0242] 给定，其中H(p)表示二进制熵函数3，且

[0243] $$H'(p)=\log_2\left(\frac{1-p}{p}\right) \quad (B7)$$

[0244] 为该率的一阶导数。我们进一步假定区间(0,0.5]中机率的分布由f(p)给定，其中$\int_0^{0.5} f(p)\,dp=1$。随后，具有相应代表性机率$\{p_{I_k}\}$的K个区间$\{I_k\}$的给定集合的以每二进制位计的预期率可写为

[0245] $$R=R(\{I_k\},\{p_{I_k}\})=\sum_{k=0}^{K-1}\left(\int_{p_k}^{p_{k+1}} R(p,p_{I_k})\,f(p)\,dp\right) \quad (B8)$$

[0246] 在k=0,...,K-1情况下的关于任何代表性机率p_{I_k}的一阶偏导数由下式给定：

[0247] $$\frac{\partial}{\partial p_{I_k}}R=\frac{p_{I_k}\int_{p_k}^{p_{k+1}}f(p)\,dp-\int_{p_k}^{p_{k+1}}p\,f(p)\,dp}{p_{I_k}(1-p_{I_k})\ln 2} \qquad (B9)$$

[0248] 方程式$\frac{\partial}{\partial p_{I_k}}R=0$对于定义域$I_k$内部的代表性机率$p_{I_k}$具有单一解。

[0249] $$p_{I_k}^{*}=\frac{\int_{p_k}^{p_{k+1}}p\,f(p)\,dp}{\int_{p_k}^{p_{k+1}}f(p)\,dp} \qquad (B10)$$

[0250] 此解的二阶偏导数

[0251] $$\frac{\partial^2}{\partial p_{I_k}^2}R(p_{I_k}^{*})=\frac{\int_{p_k}^{p_{k+1}}f(p)\,dp}{p_{I_k}^{*}(1-p_{I_k}^{*})\ln 2} \qquad (B11)$$

[0252] 始终大于零，若

[0253] $$\int_{p_k}^{p_{k+1}}f(p)\,dp>0 \qquad (B12)$$

[0254] 因此，若满足条件B12，则在方程式B10中给定的值$p_{I_k}^{*}$为区间I_k的代表性机率，该代表性机率最小化在给定区间界限p_k及p_{k+1}情况下的预期总率R。否则，没有任何二进制投射至区间I_k且代表性机率$p_{I_k}\in I_k$可经任意选择而对总率R不具有任何影响；但由于区间I_k将不用于熵编码，故应避免此配置。

[0255] 为找到最佳区间界限的条件，我们关于区间界限p_k研究预期总率R的一阶导数，其中k＝1，...，K-1。若针对所有$p\in[p_{I_{k-1}},p_{I_k})$，f(p)>0，则方程式$\frac{\partial}{\partial p_k}R=0$对于定义域$[p_{I_{k-1}},p_{I_k})$内部的区间界限$p_k$具有单一解，

[0256] $$p_k^{*}=\frac{H(p_{I_k})-p_{I_k}H'(p_{I_k})-H(p_{I_{k-1}})+p_{I_{k-1}}H'(p_{I_{k-1}})}{H'(p_{I_{k-1}})-H'(p_{I_k})} \qquad (B13)$$

[0257] 且此解的二阶偏导数

[0258] $$\frac{\partial^2}{\partial p_k^2}R(p_k^{*})=f(p_k^{*})\left(H'(p_{I_{k-1}})-H'(p_{I_k})\right) \qquad (B14)$$

[0259] 始终大于零，以便p_k^{*}为区间界限$p_k\in[p_{I_{k-1}},p_{I_k})$，该区间界限最小化在给定区间代表$p_{I_{k-1}}$及$p_{I_k}$情况下的预期总率R。若存在机率$p\in[p_{I_{k-1}},p_{I_k})$，其中f(p)＝0，则方程式$\frac{\partial}{\partial p_k}R=0$具有多个解，但方程式B13中给定的$p_k^{*}$仍为最佳的，即使进一步最佳解可能存在。

[0260] 给定区间数目K及机率分布f(p)，可经由求解经受针对k＝0，...，K-1的条件B12的方程式B10及B13给定的方程式体系，来获取最小化预期总率R的区间界限p_k(其中k＝

1,...,K-1)及区间代表p_{I_k}(其中k＝0,...,K-1)。可用以下迭代算法实现此举。

[0261] 算法1：

[0262] 1)针对所有k＝0,...,K-1,以针对所有k＝0,...,K-1遵照条件B12的方式将区间(0,0.5]分割成K个任意区间$I_k=(p_k,p_{k+1}]$,其中$p_0=0$、$p_K=0.5$,且$p_k<p_{k+1}$。

[0263] 2)根据方程式B10更新代表p_{I_k},其中k＝0,...,K-1

[0264] 3)根据方程式B13更新区间界限p_k,其中k＝1,...,K-1

[0265] 4)重复先前的两个步骤,直至收敛为止

[0266] 图14图示使用所述算法的最佳区间离散化的实例。针对此实例,我们假定对于$0<p\leq 0.5$的均匀机率分布f(p)＝2且将机率区间(0,0.5]分割成K＝4个区间。可看出机率区间离散化产生二进制熵函数H(p)的分段线性近似法A(p),其中对于所有$p\in(0,0.5]$而言$A(p)\geq H(p)$。

[0267] 作为区间离散化对编码效率的影响的测度,可使用相对于熵极限

[0268] $$\bar{\rho}=\frac{R}{\int_0^{0.5}H(p)\,f(p)\,dp}-1 \qquad (B15)$$

[0269] 的预期总率增加。对于图14的特定实例而言,熵的期望值$\bar{H}=\int_0^{0.5}H(p)\,f(p)\,dp$等于1/(2ln2)每二进制位且率额外负担$\bar{\rho}$等于1.01％。表4针对区间的选定数目K,分别列出均匀机率分布的率额外负担$\bar{\rho}_{uni}$及线性增加机率分布f(p)＝8p的率额外负担$\bar{\rho}_{lin}$,其中$p\in(0,0.5]$。

[0270] 表4:均匀及线性增加机率分布的率额外负担相对机率区间的数目

[0271]

K	1	2	4	8	12	16
$\bar{\rho}_{uni}$ [%]	12.47	3.67	1.01	0.27	0.12	0.07
$\bar{\rho}_{lin}$ [%]	5.68	1.77	0.50	0.14	0.06	0.04

[0272] 此节中研究显示LPB机率区间(0,0.5]离散化成具有固定机率的少量区间(例如,8个至10个区间)对编码效率具有非常小的影响。

[0273] 机率区间的以上论述的熵编码因此使用固定机率实现个别编码器。

[0274] 在下文中,我们首先显示可如何针对固定机率设计简单码。给定该结果,我们开发算法,该算法共同地最佳化编码设计及LPB机率区间(0,0.5]的分割。

[0275] 可使用算术编码或可变长度编码执行固定机率$p=p_{I_k}$的熵编码。对于后者情况而言,以下方法似乎简单且十分有效率。

[0276] 我们考虑二进制熵编码方案,经由该二进制熵编码方案,将可变数目的二进制映射至可变长度码字组上。对于唯一可解码性而言,将码字组逆映像至二进制序列必须是唯一的。且由于我们想要设计尽可能近地接近熵极限的码,故我们将我们的考虑限制于双射映射。此双射映像可由二元树表示,其中所有叶节点与码字组相关联,如图15中所示。树边缘表示二进制事件。在图15的实例中,下边缘表示LPB二进制值,而上边缘表示MPB二进制

值。若二元树为满二元树，亦即，若每一节点为叶或者每一节点具有两个子体，则该二元树表示二进制的前缀码。每一叶节点与基于给定的LPB机率的机率p相关联。根节点具有机率$p_{root}=1$。经由将具有LPB子体的p与MPB子体的q＝1-p的相应祖是体的机率相乘，来获取所有其它节点的机率。每一叶节点L_l的特征在于自根节点至叶节点的LPB边缘的数目a_l及MPB边缘的数目b_l。对于特定LPB机率p而言，叶节点$L_l=\{a_l,b_l\}$的机率p_l等于

[0277] $$p_l = p^{a_l}(1-p)^{b_l} \qquad (B16)$$

[0278] 二元树T的特征完全在于叶节点的数目L及相关联对$\{a_l,b_l\}$，其中l＝0,...,L-1。

[0279] 给定满二元树T及LPB机率p，可经由霍夫曼算法获取码字组至叶节点的最佳分配。所得可变数目的位至可变长度码字组(V2V)的映射C的特征在于码字组的数目L，码字组的该数目与叶节点的数目及l＝0,...,L-1的元组$\{a_l,b_l,l_l\}$一致，其中l_l表示与相应叶节点$L_l=\{a_l,b_l\}$相关联的码字组长度。应注意，存在给定码字组长度$\{l_l\}$的码字组分配的多个可能性，且只要码字组表示唯一可解的前缀码，则实际的码字组分配并不重要。针对给定码C及LPB机率p的以每二进制位计的预期率R(p,C)为预期码字组长度与每码字组二进制的期望值的比率

[0280] $$R(p,C)=\frac{\sum_{l=0}^{L-1}p_l\,l_l}{\sum_{l=0}^{L-1}p_l(a_l+b_l)}=\frac{\sum_{l=0}^{L-1}p^{a_l}(1-p)^{b_l}l_l}{\sum_{l=0}^{L-1}p^{a_l}(1-p)^{b_l}(a_l+b_l)} \qquad (B17)$$

[0281] 编码设计通常受到如码字组的最大数目L、每码字组二进制的最大数目或最大码字组长度的因素的限制，或编码设计限于特定结构的编码(例如，用于允许经最佳化的剖析)。若我们假定给定特定应用的可用码的集合S_C，则可经由最小化预期率R(p,C)来找到特定LPB机率p的最佳码$C^*\in S_C$。

[0282] $$C^*(p)=\arg\min_{\forall C\in S_C}R(p,C) \qquad (B18)$$

[0283] 作为较快的替代性实施例，最小化亦可在二元树的给定集合S_T内进行，且对于每一树而言，考虑经由霍夫曼算法获取的仅一个V2V码C。例如，我们经由考虑所有二元树T，针对各种LPB机率p设计V2V码，该所有二元树T的叶节点的数目L小于或等于给定的最大值L_m。在图16中，在选定的最大表大小L_m的LPB机率p内绘制相对率增加$\rho(p,C^*(p))=R(p,C^*(p))/H(p)$。通常可经由允许较大的表大小来减小率增加ρ(p)。对于较大的LPB机率而言，8个至16个码字组的较小的表大小L通常足以保持率增加ρ(p)合理地较小，但对于较小的LPB机率(例如，p<0.1)而言，需要较大的表大小L。

[0284] 在先前的节中，我们考虑假定最佳码的最佳机率离散化及固定LPB机率的编码设计。但由于一般而言我们无法用有限的表大小的真实V2V码实现熵极限，故必须共同考虑编码设计及LPB机率区间(0,0.5]的分割，以获取最佳化的熵编码设计。

[0285] 对于给定区间$I_k=(p_k,p_{k+1}]$而言，若码C_k最小化该给定区间的预期率$R=\int_{p_k}^{p_{k+1}}R(p,C_k)\,f(p)\,dp$，则最佳码$C_k^*$中的给定集合$S_C$的码$C_k$

[0286] $$C_k^*=\arg\min_{\forall C_k\in S_C}\int_{p_k}^{p_{k+1}}R(p,C_k)\,f(p)\,dp \qquad (B19)$$

[0287] 对于实用的设计而言，方程式B19中整数的最小化可经由首先根据方程式B10针对区间I_k决定最佳代表性机率$p_{I_k}^*$且随后根据方程式B18针对代表性机率$p_{I_k}^*$选择给定集合S_C的最佳码C_k^*来简化，而对编码效率具有较小影响。

[0288] 可经由最小化预期总率来导出给定码的集合C_k(其中k＝0,...,K-1)情况下的最佳区间界限p_k(其中k＝1,...,K-1)。

[0289] $$R = R(\{p_k\},\{C_k\}) = \sum_{k=0}^{K-1}\left(\int_{p_k}^{p_{k+1}} R(p,C_k)\, f(p)\, dp\right) \qquad (B20)$$

[0290] 针对k＝1,..,K-1，相对于区间界限将一阶导数设定成等于零，$\frac{\partial}{\partial p_k} R = 0$，得到

[0291] $$p_k^* = p_k \quad with \quad R(p_k, C_{k-1}) = R(p_k, C_k) \qquad (B21)$$

[0292] 类似地，关于方程式B13，可以表明p_k^*始终为最佳解，但取决于机率分布f(p)，另外的最佳解可能存在。因此，在分别给定相关联码C_{k-1}及C_k的情况下，两个区间I_{k-1}及I_k之间的最佳区间界限p_k^*为函数R(p,C_{k-1})与R(p,C_k)的交点。

[0293] 因此，以下交互式算法可用于在给定机率区间的数目K、可能码的集合S_C及机率分布f(p)，其中p∈(0,0.5]的情况下，共同地导出机率区间分割及相关联码。

[0294] 算法2：

[0295] 1)使用在节3中指定的算法1，导出初始机率区间界限p_k，其中k＝0,...,K

[0296] 2)根据方程式B10导出机率区间I_k的代表p_{I_k}，其中k＝0,...,K-1

[0297] 3)根据方程式B18导出区间代表p_{I_k}的码C_k∈S_C，其中k＝0,...,K-1

[0298] 4)根据方程式B21更新区间界限p_k，其中k＝1,...,K-1

[0299] 5)重复先前的三个步骤，直至收敛为止

[0300] 算法2中的步骤2及步骤3亦可根据方程式B19，基于区间界限p_k，其中k＝0,...,K，由直接导出码C_k∈S_C，其中k＝0,...,K-1替代。此外，如节4.1中所提及的，步骤3中的最小化亦可在二元树的给定集合S_T内进行，其中对于每一二元树T而言，考虑经由霍夫曼算法获取的仅一个V2V码C_k。

[0301] 例如，我们使用算法2共同导出分割成K＝12个机率区间及相应的V2V码。此后，该算法的步骤3中的最小化以二元树的给定集合S_T内等效的最小化来替代，其中每一树T的经评估码C经由霍夫曼算法获取。我们考虑具有叶节点的最大数目L_m＝65的树T，因此考虑具有高达65个表项的码C。已在最小化中评估具有高达16个叶节点的所有二元树T；对于具有多于16个叶节点的树而言，在给定具有更小数目叶节点的树的更佳结果的情况下，我们采用次最佳的搜寻。

[0302] 在图17中，在LPB机率p内绘制编码设计实例的相对于熵极限的预期率增加ΔR(p)＝R(p)-H(p)。作为比较，我们亦针对理论上最佳的机率区间离散化(如节3中揭示的)及具有图内部额外限制$p_{I_{K-1}} = 0.5$的理论上最佳的机率离散化，绘制预期率增加ΔR。可看出共

同的机率区间离散化及V2V编码设计导致区间界限移动(区间界限p_k由ΔR(p)曲线的局部最大值给定,其中k=1,...,K-1)。当假定均匀机率分布f(p)时,具有真实的V2V码的设计实例的相对于熵极限的相对预期总率增加为$\bar{\rho}=0.24\%$。理论上最佳的机率区间离散化及具有额外限制$p_{I_{K-1}}=0.5$的理论上最佳的机率离散化的相应的相对率增加分别为$\bar{\rho}=0.12\%$及$\bar{\rho}=0.13\%$。

[0303] 可如下执行码字组终止。当编码符号的有限序列$\{s_0,...,s_{N-1}\}$时,K个二进制编码器中的每一二进制编码器处理编码二进制的有限序列$\boldsymbol{b}_k^c=\{b_0^c,...,b_{B_k-1}^c\}_k$,其中k=0,...,K-1。且已确保,对于K个二进制编码器中的每一二进制编码器而言,所有编码二进制序列$\boldsymbol{b}_k^c=\{b_0^c,...,b_{B_k-1}^c\}_k$可在给定码字组或码字组的序列$\mathbf{c}_k(\mathbf{b}_k^c)$的情况下重建。

[0304] 当采用算术编码时,编码二进制的序列的算术码字组必须以在给定码字组的情况下可解码所有编码二进制的方式终止。对于以上所述V2V码而言,序列$\mathbf{b}_k^c$末端处的二进制可能不表示与码字组相关联的二进制序列。在此情况下,可写入含有剩余二进制序列作为前缀的任何码字组。若选择具有最小长度的相应的码字组(或该码字组中的一个码字组),则可最小化额外负担。在解码器侧,废除在二进制序列的末端处额外读取的二进制,在给定位串流语法及二进制化方案的情况下可识别该二进制。

[0305] 下文呈现简单的编码设计实例。出于说明的目的,我们考虑具有三个字母的源{s}及$p_s(a_0)=0.7$、$p_s(a_1)=0.18$及$p_s(a_2)=0.12$的固定相关联机率的简单实例。可将相应的三元选择树转换成图18中所示的满二元树。

[0306] 在表5中给出图18中的满二元树的二进制化。将三进制符号pmfp_s转换成两个二进制$\mathrm{pmf}\,p_{b_0}=(0.7,0.3)$及$p_{b_1}=(0.6,0.4)$。对于位串流中的每一符号s而言,存在二进制$b_0$。当$b_0$等于0时,$b_1$亦存在。注意到表2中给定的二进制化与源s的最佳的单一字母霍夫曼码一致。

[0307] 表5:三字母源的二进制化。第一二进制的LPB机率p_{LPB}为0.3且第二二进制的LPB机率p_{LPB}为0.4。

[0308]

符号a_i	机率$p(a_i)$	二进制b_0	二进制b_1
a_0	0.7	1	
a_1	0.18	0	1
a_2	0.12	0	0
LPB机率	$p_{LPB}=p(b_j=0)$	0.3	0.4

[0309] 源s的熵为

[0310] H(0.7,0.18,0.12)=H(0.7,0.3)+0.3H(0.6,0.4)

[0311] =1.1726 bit/symbol(B22)

[0312] 单一字母霍夫曼码的平均码字长度给定为对应于$\rho_{HC}=0.1274$位/符号的冗余度或10.87%预期率额外负担的

[0313] $$\bar{\ell}_{HC}=\sum_{i=0}^{M-1}p_i\ell_i^{HC}=1.3\ bit/symbol \qquad (B23)$$

[0314] 对于具有固定pmf的特定二进制化实例而言，由于针对两个二进制，LPB值b^j_{LPB}等于0，故二进制b_0及b_1已表示编码二进制。LPB机率的分布f(s)是离散的，其中除p＝0.3及p＝0.4之外，f(p)＝0。因此，最佳机率离散化产生具有代表$p_{I_0}=0.3$及$p_{I_1}=0.4$的K＝2个区间。可在[0.3,0.4)中任意选择该区间之间的区间界限p_1。

[0315] 为编码该源，将源符号的序列二进制化成二进制的序列。针对每一源符号传输二进制b_0。仅当b_0＝0时传输二进制b_1。分别用$p_{I_0}=0.3$及$p_{I_1}=0.4$的恒定的LPB机率单独编码二进制b_0及b_1。

[0316] 可经由简单的V2V映射来实现具有固定机率的二进制字母表的高效编码。分别在表6及表7中给出具有LPB机率p_{LPB}＝0.3及p_{LPB}＝0.4的较小的编码表的V2V映像的实例。p_{LPB}＝0.3的V2V映射产生0.0069bit/bin或0.788％的冗余度。对于p_{LPB}＝0.4的LPB机率而言，冗余度为0.0053bit/bin或0.548％。

[0317] 表6：p_{LPB}＝0.3的LPB机率的二元树及码。码的冗余度为0.788％。

[0318]

二元树	机率	码
’11′	0.7^2＝0.49	’1′
’01′	0.7•0.3＝0.21	’01′
’0′	0.3	’00′

[0319] 表7：p_{LPB}＝0.4的LPB机率的二元树及码。此码的冗余度为0.548％。

[0320]

二元树	机率	码树
’111′	0.6^3＝0.216	’11′
’110′	0.6^2•0.4＝0.144	’001′
’10′	0.6•0.4＝0.24	’11′
’01′	0.4•0.6＝0.24	’01′
’00′	0.4^2＝0.16	’000′

[0321] 经由新的编码方法产生的总预期率为

[0322]
$$\bar{\ell}_{NC}=\bar{\ell}_{b_0}+0.3\cdot\bar{\ell}_{b_1}$$
$$=1.181\ bit/symbol \quad (B24)$$

[0323] 总冗余度为相对于熵极限0.73％，该总冗余度表示与单一字母霍夫曼码相比显著的改良。

[0324] 可说明的是，可经由产生执行长度码来获取相似的编码效率改良。针对以上实例，我们可经由考虑高达两个符号的执行，来建构最大机率符号的执行长度码。事件{a_0a_0，a_0a_1，a_0a_2，a_1，a_2}中的每一事件将与单独的码字组相关联。此码产生相对于熵极限1.34％的冗余度。实际上，V2V码可视为二进制符号的执行长度码的一般化(表3中的V2V码确实有效地表示执行长度码)。对于具有固定机率的单一符号字母表而言，亦可经由产生码来实现关于本方法的相似的编码效率，该码将可变数目的源符号映像至可变长度码字组。本方法的主要优点为在将具有固定或适应性机率估计的任意的源符号序列映像至用固定LPB机率操作的少量简单二进制编码器时该方法的灵活性。

[0325] 接下来考虑如何实现唯一可解码性。

[0326] 使用本熵编码方案，源符号的序列$s=\{s_0,\ldots,s_{N-1}\}$的编码由以下三个基本步骤组成：

[0327] • 符号二进制化$b=\{b_0,\ldots,b_{B-1}\}$产生二进制的序列$b=\{b_0,\ldots,b_{B-1}\}$

[0328] • 二进制的序列转换成编码二进制的序列$\boldsymbol{b}^c=\{b_0^c,\ldots,b_{B-1}^c\}=\gamma_c(\mathbf{b})$

[0329] • 使用机率区间离散化及K个固定二进制编码器二进制熵编码编码二进制的序列$\boldsymbol{b}^c=\{b_0^c,\ldots,b_{B-1}^c\}$

[0330] 若编码二进制的序列$\boldsymbol{b}^c=\{b_0^c,\ldots,b_{B-1}^c\}$为唯一可解码的且映像$\gamma_b$及$\gamma_c$为可逆的，则符号序列$s=\{s_0,\ldots,s_{N-1}\}$为唯一可解码的。

[0331] 让γ_e通知编码器将一或更多编码二进制的序列$\boldsymbol{b}^c=\{b_0^c,\ldots\}$映射至一或更多码字组的序列$c(b^c)=\{c_0,\cdots\}$上。

[0332] $$c(b^c)=\gamma_e(b^c) \qquad (B25)$$

[0333] 对于给定码字组的序列$c(b^c)$情况下的编码二进制的序列b^c的唯一可解码性而言，编码器映像γ_e必须具有将唯一码字组$c(b^c)$分配予编码二进制的每一可能序列b^c的性质：

[0334] $$\forall \boldsymbol{b}_i^c,\boldsymbol{b}_j^c \quad \boldsymbol{b}_i^c\neq\boldsymbol{b}_j^c \quad\rightarrow\quad \boldsymbol{c}(\boldsymbol{b}_i^c)\quad\neq\boldsymbol{c}(\boldsymbol{b}_j^c) \qquad (B26)$$

[0335] 当使用算术码或前缀码时，始终满足此性质。由于V2V码表示可变数目的二进制的前缀码，故对于节4.1中描述的V2V码(包括节4.3中描述的码字组终止)而言尤其满足此性质。

[0336] 然而，在本熵编码方法中，将编码二进制的序列b^c分割成K个子序列$\boldsymbol{b}_k^c$，其中$k=0,\ldots,K-1$，

[0337] $$\{b_0^c,\cdots,b_{K-1}^c\}=\gamma_p(b^c) \qquad (B27)$$

[0338] 且对于子序列$\boldsymbol{b}_k^c$中的每一子序列，使用特定编码器映像γ_e^k分配码字组的序列$c_k(b_k^c)$。因此，必须延伸唯一可解码性的条件。若在给定相应码字组$c_k(b_k^c)$的情况下编码二进制的每一子序列b_k^c为唯一可解码的，且分割规则γ_p对编码器已知，则在给定码字组$c_k(b_k^c)$的K个序列，其中$k=0,\ldots,K-1$的情况下，编码二进制的序列b^c为唯一可解码的。分割规则γ_p由LPB机率区间离散化$\{I_k\}$及与编码二进制b_j^c相关联的LPB机率p_{LPB}^j给定，其中$j=0,\ldots,B-1$。因此，LPB机率区间离散化$\{I_k\}$必须在解码器侧已知，且必须在编码器及解码器侧以相同方式导出每一编码二进制b_j^c的LPB机率p_{LPB}^j，其中$j=0,\ldots,B-1$。

[0339] 为将二进制的序列映射γ_c至编码二进制的序列c上，将每一单一b_j经由二进制映射$b_j^c=\gamma_c^j(b_j)=b_j\oplus b_{LPB}^j$来转换，其中$j=0,\ldots,B-1$。在解码器侧，二进制序列可经由二进制映射来导出，

[0340] $$b_j=(\gamma_c^j)^{-1}(b_j^c)=b_j^c\oplus b_{LPB}^j \qquad (B28)$$

[0341] 其中$j=0,\ldots,B-1$。由于

[0342] $$b_j^c\oplus b_{LPB}^j=b_j\oplus b_{LPB}^j\oplus b_{LPB}^j=b_j\oplus 0=b_j \qquad (B29)$$

[0343] 故若在编码器及解码器侧以相同方式导出每一二进制b_j的LPB值b^j_{LPB}，则该映射$(\gamma^j_c)^{-1}$表示相应的编码器映像γ^j_c的逆向，因此，二进制序列b转换γ_b成编码二进制的序列b^c是可逆的。

[0344] 最终，我们研究二进制化$b=\gamma_b(s)$的可逆性，经由该可逆性将每一符号s_i映射至二进制序列$\boldsymbol{b}^i=\gamma^i_b(s_i)$上，其中$i=0,\ldots,N-1$。若二进制化映射$\gamma^i_b$将不同的二进制序列$\boldsymbol{b}^j_m$分配予符号$s_i$的字母表$A_i$的每一字母$a^i_m$，则在给定相应的二进制序列$b^i$情况下，可唯一解码符号$s_i$。然而，由于将二进制序列$b=\{b_0,\ldots,b_{B-1}\}$分割成对应于符号$s_i$的二进制序列$b^i$对解码器并非已知，其中$i=0,\ldots,N-1$，故此条件并不充分。当对于每一符号$s_i$而言，与相应的字母表$A_i$的字母$a^i_m$相关联的二进制序列$\boldsymbol{b}^j_m$形成前缀码，且每一符号$s_i$的二进制化映射$\gamma^i_b$在解码器侧已知，其中$i=0,\ldots,N-1$时，给定充分条件。

[0345] 本熵编码方法的唯一可解码性的条件可概述如下：

[0346] • 二进制化映射γ^i_b表示前缀码且对解码器已知(按符号编码次序)

[0347] • 在编码器及解码器侧以相同方式导出所有二进制b_j的机率模型(b^j_{LPB},p^j_{LPB})

[0348] • 将LPB机率区间(0,0.5]分割成K个区间I_k对对解码器已知，其中$k=0,\ldots,K-1$

[0349] • 每一机率区间I_k的映射γ^k_e表示唯一可解码，其中$k=0,\ldots,K-1$

[0350] 在下文中，我们更详细地描述整体编码器及解码器设计的实例。我们集中于编码方案，在该编码方案中，在编码器及解码器侧直接估计二进制的机率模型$\{b_{LPB},p_{LPB}\}$且K个二进制编码器使用以上所述的V2V映射。每一源符号s应与符号类别c_s相关联，该符号类别c_s决定符号的类型，符号的该类型包括该符号的值范围。符号及相关联符号类别的次序应经由语法来给定，假定该语法在编码器及解码器侧已知。

[0351] 示例性PIPE编码器及PIPE解码器设计的方块图图示于图19中。在编码器侧，将具有相关联符号类别c_s的符号s馈送至二进制编码器，该二进制编码器将每一符号s转换成二进制的序列$\boldsymbol{b}_s=\gamma^{c_s}_b(s)$。

[0352] 基于符号类别c_s，决定所使用的二进制化方案$\gamma^{c_s}_b$。此外，二进制编码器使二进制序列s的每一二进制b与机率模型指示c_b相关联，该机率模型指示c_b指定用于编码二进制b的机率模型。可基于符号类别c_s、二进制序列s内部当前二进制的二进制数目，及/或已编码二进制及符号的值导出机率模型指示c_b。

[0353] 机率估计器及分配器维持多个机率模型，该多个机率模型的特征在于值对$\{b_{LPB},p_{LPB}\}$。该机率估计器及分配器自二进制编码器接收二进制b及相关联机率模型指示c_b且将LPB值b_{LPB}及所指示的机率模型的LPB机率p_{LPB}分别转发至编码二进制导出器及机率量化器。此举的后，使用所接收的二进制b的值更新相应的机率模型$\{b_{LPB},p_{LPB}\}$。

[0354] 编码二进制导出器分别自二进制编码器及机率估计器及分配器接收二进制b及相关联LPB值b_{LPB}且将编码二进制b^c发送至机率量化器，该编码二进制b^c经由$b^c=b\oplus b_{LPB}$导出。机率量化器将每一编码二进制b^c转发至K个二进制编码器中的一个二进制编码器。机率

量化器含有关于LPB机率区间量化$\{I_k\}$的信息。将LPB机率p_{LPB}与区间界限$\{p_k\}$相比较，该LPB机率p_{LPB}与编码二进制b_c相关联且自机率估计器及分配器接收，且导出机率区间标示k，对于该机率区间标示k，$p_{LPB}\in I_k$。随后，将编码二进制b^c转发至相关联二进制编码器。

[0355] K个二进制编码器中的每一二进制编码器由二进制缓冲器及二进制编码器组成。二进制缓冲器自机率量化器接收编码二进制b^c且按编码次序储存该编码二进制b^c。二进制编码器实施特定V2V映射，且二进制编码器将二进制缓冲器中的二进制序列同与码字组相关联的二进制序列相比较。若二进制缓冲器中的二进制序列等于彼等二进制序列中的一个二进制序列，则二进制编码器自二进制缓冲器移除二进制序列$\{b^c\}$且将相关联码字组($\{b^c\}$)写入至相应的码字组串流。在符号序列的编码程序的末端处，针对所有二进制编码器，如节4.3中所述写入终止码字组，该所有二进制编码器的二进制缓冲器不是清空的。

[0356] 可单独传输、包化或储存K个所得码字组串流，或出于传输或储存的目的可将该码字组串流交错(比较节6.2)。

[0357] 在解码器侧，由二进制解码器及二进制缓冲器组成的K个二进制解码器中的每一二进制解码器接收一个码字组串流。二进制解码器自码字组串流读取码字组($\{b^c\}$)且将相关联二进制序列$\{b^c\}$按编码次序插入至二进制缓冲器中。

[0358] 符号序列的解码经由下层语法驱动。将符号s的请求与符号类别c_s一起发送至二进制编码器。二进制编码器将该符号请求转换成二进制的请求。二进制的请求与以与在编码器中的方式相同的方式导出的机率模型指示c_b相关联，且将二进制的请求发送至机率估计器及分配器。类似于编码器侧处的机率估计器及分配器的对应物来操作该机率估计器及分配器。基于机率模型指示c_b，机率估计器及分配器识别机率模型且将该机率模型的LPB值b_{LPB}及LPB机率p_{LPB}分别转发至二进制导出器及机率量化器。

[0359] 机率量化器以与在编码器侧决定二进制编码器的方式相同的方式，基于LPB机率p_{LPB}决定K个二进制解码器中的一个二进制解码器，机率量化器自相应的二进制缓冲器按编码次序移除第一编码二进制b^c，且机率量化器将该第一编码二进制b^c转发至二进制导出器。二进制导出器分别自机率量化器及机率估计器及分配器接收编码二进制b_c及相关联LPB值b_{LPB}，且二进制导出器决定二进制值$b=b^c\oplus b_{LPB}$。作为对二进制编码器发送的二进制请求的最终回应，二进制导出器将经解码二进制值b发送至二进制编码器及机率估计器及分配器。

[0360] 在机率估计器及分配器中，经解码二进制值b用以用与编码器侧的方式相同的方式更新机率模型$\{b_{LPB},p_{LPB}\}$，该机率模型$\{b_{LPB},p_{LPB}\}$经由相关联值c_b选定。最终，二进制编码器向已针对符号请求接收的二进制序列s添加所接收的二进制b，且二进制编码器经由二进制化方案$\gamma_b^{c_s}$将此二进制序列$_s$同与符号值相关联的二进制序列相比较。若二进制序列$_s$与彼等二进制序列中的一个二进制序列匹配，则输出相应的经解码符号s作为对符号请求的最终响应。否则，二进制编码器发送另外的二进制请求，直至解码符号为s止。

[0361] 若未接收经由语法驱动的另外的符号请求，则终止符号序列的解码。废除可能在熵解码程序的末端处的二进制缓冲器中含有的编码二进制b^c(作为终止码字组的结果)。

[0362] 在关于图3至图13描述图1A-图1C及图2A和图2B中的PIPE编码器及PIPE解码器的某些实施例且关于图14至图19提供大体有关PIPE编码的数学背景的后，关于图20至图24，

描述熵编码及解码设备的更详细的实施例。图22至图24的以下实施例不仅交错彼此之中的经PIPE编码位串流，而且总共交错VLC位串流及经PIPE编码的位串流。与图22至图24的实施例相比较，具有经PIPE编码位串流的交错的PIPE编码器及PIPE解码器的实施例，亦即图7至图13的实施例仅提供经PIPE编码位串流的单独交错。如以上已提及的，经由分别使用另一交错器134/去交错器228对(参见图1A-图1C及图2A和图2B)，诸如经由使用图5及图6中所示的位串流交错，甚至该实施例可充当实现完全经交错位串流的基础。然而，接下来描述的实施例在VLC位串流以及经PIPE编码位串流两者的后立刻执行交错，其中分别使用(换言的)一阶段交错器128/去交错器230。

[0363] 在详细描述经VLC及PIPE编码符号在位串流内部交错的实施例之前，为了实现复杂性与编码效率之间更适合的取舍，关于图20及图21描述不具有交错的实施例的基本结构。

[0364] 图1A的熵编码设备的结构图标于图20中。熵编码设备将对应于图1A-图1C及图2A和图2B的经VLC编码源符号与经PIPE编码源符号的(亦即106及218)组合的源符号1a的串流分别转换成两个或两个以上部分位串流12、12a的集合，其中位串流12a对应于图1A-图1C及图2A和图2B的位串流112及206。

[0365] 如以上已批注的，每一源符号1a可具有与该每一源符号1a相关联的指示，该指示指定是否使用VLC编码器22a内的标准VLC码来编码源符号或关于是否将使用PIPE编码概念编码源符号，该VLC编码器22a对应于图1A-图1C中的VLC编码器102。如以上已关于图1A-图1C及图2A和图2B所描述的，可能不将此指示显性传输至解码侧。相反地，该相关联指示可源自源符号本身的类型或类别。

[0366] 使用标准VLC码来编码经VLC编码符号1b，该标准VLC码又可使用VLC编码器22a而取决于刚刚提及的符号类别或符号类型。将相应的码字组11a写入至不同的部分位串流12a。使用以上关于图1A-图1C及图3所述的PIPE编码来编码非经VLC编码符号1，例如，在此的后获取多个部分位串流12。源符号1a中的一些源符号1a可能已经以二进制化由以上已关于图1A提及的两个部分组成的方式二进制化。该部分中的一个部分可使用PIPE方法来编码，且可将该部分中的一个部分写入至相应的部分位串流12。二进制序列的另一部分可使用标准VLC码来编码，且可将二进制序列的另一部分写入至相应的部分位串流12a。

[0367] 适于图20的实施例的基本熵解码设备图标于图21中。

[0368] 解码器基本上执行图20的编码器的逆向操作，以便自两个或两个以上部分位串流的集合(24、24a)解码源符号27、27a的先前编码的序列。解码器包括两个不同处理流程：数据请求流及数据流，该数据请求流复制编码器的数据流，该数据流表示编码器数据流的逆向。在图21的说明中，虚线箭头表示资料请求流，而实线箭头表示资料流。解码器的构建块基本上复制编码器的构建块，但实施逆向操作。

[0369] 在本发明的较佳实施例中，每一符号请求13a与指示相关联，该指示指定是使用标准VLC码还是使用PIPE编码概念来编码源符号。如以上已关于图20提及的，此指示可源自剖析规则或由源符号本身表示的语法元素的语法。举例而言，关于图1A-图1C及图2A和图2B，已描述不同语法元素类型可与不同编码方案相关联，该不同编码方案亦即VLC编码或PIPE编码。编码方案可应用二进制化的不同部分或更大体而言，语法元素的其它符号化。若符号经VLC编码，则将请求传递至VLC解码器22a，且自不同的部分位串流24a读取VCL码字组23a。

输出相应的解码符号27a。例如，若使用PIPE来编码符号，自以上关于图4所述的部分位串流的集合24解码符号27。

[0370] 在本发明的另一较佳实施例中，以二进制化由两个部分组成的方式来二进制化源符号中的一些源符号。使用PIPE方法编码的该部分中的一个部分相应地自相关联部分位串流24解码。且二进制序列的另一部分使用标准VLC码来编码且使用VLC解码器22a来解码，该VLC解码器22a自不同的部分位串流24a读取相应的码字组23a。

[0371] 部分位元串流(经VLC编码及经PIPE编码)的传輸及多工

[0372] 可单独传输经由PIPE编码器产生的部分位串流12、12a，或可将该部分位串流12、12a多任务成单一位串流，或可将部分位串流的码字组交错于单一位串流中。

[0373] 在本发明的较佳实施例中，将大量数据的每一部分位串流写入至一个数据封包。该大量数据可为源符号的任意集合，诸如，静止图像、视讯序列的字段或讯框、静止图像的片段、视讯序列的字段或讯框的片段或音讯取样的讯框等。

[0374] 在本发明的另一较佳实施例中，将大量数据的部分位串流12、12a中的两个或两个以上部分位串流12、12a或大量数据的所有部分位串流多任务成一个数据封包。含有经多任务部分位串流的数据封包的结构可如图5中所示。

[0375] 数据封包300由标头及每一部分位串流的数据的一个分区组成(对于所考虑量的数据而言)。数据封包的标头301含有将数据封包(的剩余部分)分割成位串流数据302的区段的指示。除对于分割的指示之外，标头可含有额外的信息。在本发明的较佳实施例中，对于分割数据封包的指示为在位或字节的单元或多个位或多个字节中数据区段开始的位置。在本发明的较佳实施例中，相对于数据封包的开始或相对于标头的末端或相对于先前的数据封包的开始，将数据区段开始的位置编码为数据封包的标头中的绝对值。在本发明的另一较佳实施例中，将数据区段开始的位置区别编码，亦即，仅将数据区段的实际开始与对于数据区段开始的预测之间的差异编码。基于已知或已传输的信息，诸如，数据封包的总大小、标头的大小、数据封包中数据区段的数目、居先数据区段开始的位置，可获得该预测。在本发明的较佳实施例中，不将第一数据封包开始的位置编码，而是基于数据封包标头的大小推断第一数据封包开始的该位置。在解码器侧，经传输的分割指示用于获得数据区段的开始。数据区段随后用作部分位串流12、12a，且将该数据区段中含有的数据按顺序的次序馈送至相应的二进制解码器及VLC解码器中。

[0376] 码字組(VLC及PIPE码字組)的交错

[0377] 对于一些应用而言，一个数据封包中部分位串流(针对大量源符号)的以上描述的多任务可具有以下缺点：一方面，对于小数据封包而言，发出分割讯号所需要的边信息的位的数目可相对于部分位串流中的实际数据而变得显著，此状况最终降低编码效率。另一方面，多任务可能不适合于需要低迟延的应用(例如，对于视讯会议应用而言)。由于之前并不知道分区开始的位置，故使用所述多任务，编码器无法在完全产生部分位串流之前开始传输数据封包。此外，一般而言，解码器必须等待，直至该解码器在该解码器可开始解码数据封包之前接收最后的数据区段的开始为止。对于作为视讯会议系统的应用而言，该延迟可合计为若干视讯图像的系统的额外的总延迟(特定言的针对接近传输位率且针对几乎需要两个图像之间的时间区间以编码/解码图像的编码器/解码器)，该额外的总延迟对于该应用而言十分关键。为了克服某些应用的缺点，可以将经由两个或两个以上二进制编码器及

VLC编码器产生的码字组交错至单一位串流中的方式配置本发明的较佳实施例的编码器。可将具有经交错码字组的位串流直接发送至解码器(当忽略较小的缓冲延迟时,参见下文)。在解码器侧,两个或两个以上二进制解码器及VLC解码器自位串流按解码次序直接读取码字组;解码可以首先接收的位开始。此外,发出部分位串流的多任务(或交错)讯号不需要任何边信息。

[0378] 具有码字组交错的编码器的基本结构图标于图22中。二进制编码器10及VLC编码器10a不将码字组直接写入至部分位串流,而是二进制编码器10及VLC编码器10a与单一码字组缓冲器29连接,自该单一码字组缓冲器29,将码字组按编码次序写入至位串流34。二进制编码器10将对于一或更多新的码字组缓冲项28的请求发送至码字组缓冲器29,且二进制编码器10随后将码字组30发送至码字组缓冲器29,该码字组30储存于经保留的缓冲项中。VLC编码器10a将VLC码字组30a直接写入至码字组缓冲器29。码字组缓冲器29的(一般而言可变长度)码字组31经由码字组写入器32来存取,该码字组写入器32将相应的位33写入至所产生的位串流34。码字组缓冲器29作为先进先出缓冲器操作;较早保留的码字组项被较早写入至位串流。

[0379] 在本发明的较佳实施例中,如下操作码字组缓冲器。若将新的二进制7发送至特定二进制缓冲器8,且该二进制缓冲器中已储存二进制的数目为零,且当前不存在任何码字组保留于二进制编码器的码字组缓冲器中,该二进制编码器与该特定二进制缓冲器连接,则经连接的二进制编码器10将请求发送至该码字组缓冲器,经由该码字组缓冲器,一或更多码字组项保留于该特定二进制编码器10的码字组缓冲器29中。码字组项可具有可变数目的位;通常经由相应的二进制编码器的最大码字组大小给定缓冲项中位的数目的上临界。经由二进制编码器产生的一或更多接下来的码字组(针对该一或更多接下来的码字组已保留一或更多码字组项)储存于码字组缓冲器的经保留的一或更多项中。若特定二进制编码器的码字组缓冲器中的所有经保留缓冲项填充有码字组,且将下一个二进制发送至与该特定二进制编码器连接的二进制缓冲器,则一或更多新的码字组保留于该特定二进制编码器的码字组缓冲器中等。VLC编码器10a将VLC码字组30a直接写入至码字组缓冲器29的下一个自由项,亦即,对于VLC编码器而言,立刻执行码字组保留及码字组的写入。码字组缓冲器29以某一方式表示先进先出缓冲器。缓冲项按顺序的次序保留。较早保留的相应缓冲项所针对的码字组被较早地写入至位串流。码字组写入器32连续地或者在将码字组30写入至码字组缓冲器29的后检查码字组缓冲器29的状态。若第一缓冲项含有完整的码字组(亦即,该缓冲项未经保留,而是该缓冲项包括码字组),则自码字组缓冲器20移除相应的码字组31及相应的缓冲项,且将码字组的位33写入至位串流。重复此程序,直至第一缓冲项不含有码字组(亦即,该码字组为经保留或自由的)为止。在解码程序的末端,亦即,若已处理考虑量的数据的所有源符号,则必须清除码字组缓冲器。对于彼清除程序而言,针对每一二进制缓冲器/二进制编码器应用下文作为第一步骤:若二进制缓冲器确实含有二进制,则添加具有特定或任意值的二进制,直至所得二进制序列表示与码字组相关联的二进制序列为止(如以上所批注,添加二进制的一个较佳方式为,添加产生与二进制序列相关联的最短可能码字组的该二进制值或该二进制值中的一个二进制值,该二进制序列含有二进制缓冲器的原有内容作为前缀),随后将该码字组写入至相应的二进制编码器的下一个经保留的缓冲项且清空(相应的)二进制缓冲器。若针对一或更多二进制编码器已保留多于一个缓冲项,则码

字组缓冲器可仍含有经保留的码字组项。在彼状况下，该码字组项填充有相应的二进制编码器的任意但有效的码字组。在本发明的较佳实施例中，插入最短有效码字组或该最短有效码字组中的一个最短有效码字组(若存在多个)。VLC编码器不需要任何终止。最终，将码字组缓冲器中所有剩余码字组写入至位串流。

[0380] 码字组缓冲器的状态的两个实例图标于图23A和图23B中。在实例(a)中，码字组缓冲器含有填充有码字组的4个项(该4个项中的两个项为VLC项)及3个经保留项。此外，标记下一个自由缓冲项。第一项填充有码字组(亦即，二进制编码器2刚刚将码字组写入至先前保留的项)。在下一个步骤中，将自码字组缓冲器移除此码字组，且将此码字组写入至位串流。由于仅保留第一缓冲项，但无码字组写入至此项，故随后，二进制编码器3的首先保留的码字组为第一缓冲项，但无法自码字组缓冲器移除此项。在实例(b)中，码字组缓冲器含有填充有码字组的4个项(该4个项中的一个项为VLC缓冲项)及4个经保留项。将第一项标记为经保留，因此码字组写入器无法将码字组写入至位串流。尽管在码字组缓冲器中含有4个码字组，但码字组写入器必须等待，直至将码字组写入至二进制编码器3的首先保留的缓冲项为止。注意到必须按保留码字组的次序写入该码字组，以能够使解码器侧的程序逆向(参见下文)。且进一步注意到由于立刻执行码字组的保留及写入，故VLC缓冲项始终为完整的。

[0381] 具有码字组交错的解码器的基本结构图标于图24中。二进制解码器22及VLC解码器2a不直接自单独的部分位串流读取码字组，而是二进制解码器22及VLC解码器2a连接至位缓冲器38，自该位缓冲器38按编码次序读取码字组37、37a。应注意，由于亦可自位串流直接读取码字组，故未必需要位缓冲器38。位缓冲器38主要包括于处理链的明显单独的不同态样的说明中。将具有经交错码字组的位串流40的位39顺序插入至位缓冲器38中，该位缓冲器38表示先进先出缓冲器。若特定二进制解码器22接收一或更多二进制序列35的请求，则二进制解码器22自位缓冲器38经由位请求36读取一或更多码字组37。解码器可瞬时解码源符号。类似地，若VLC解码器22a接收新的符号的请求19a，则VLC解码器22a自位缓冲器38读取相应的VLC码字组37a且返回经解码符号27a。注意到，编码器(如以上所述)必须经由适当地操作码字组缓冲器来确保按与经由二进制解码器请求码字组的次序相同的次序将该码字组写入至位串流。在解码器处，整个解码程序经由源符号的请求来触发。作为经由特定二进制编码器保留在编码器侧的码字组的数目与经由相应的二进制解码器读取的码字组的数目的参数必须相同。

[0382] 具有低延迟限制的可变长度码字組的交错

[0383] 所描述的码字组交错不需要将任何分割信息作为边信息发送。且由于码字组交错于位串流中，故延迟一般较小。然而，并不保证遵照特定延迟限制(例如，由储存于码字组缓冲器中的位的最大数目指定的特定延迟限制)。此外，码字组缓冲器所需要的缓冲区大小可在理论上变得十分大。在考虑第23(b)图中的实例时，有可能没有另外的二进制发送至二进制缓冲器3，因此二进制编码器3将不将任何新的码字组发送至码字组缓冲器，直至应用数据封包末端处的清除程序为止。随后在可将二进制编码器1及2的所有码字组写入至位串流之前，二进制编码器1及2的所有码字组将必须等待，直至数据封包的末端为止。可经由向编码程序(且亦向随后描述的解码程序)添加另一机制来防止此缺陷。彼额外的机制的基本概念为，若与延迟有关的测度或延迟的上限(参见下文)超过指定临界，则经由清除相应的二进制缓冲器(使用与数据封包的末端相似的机制)，来填充首先保留的缓冲项。经由此机制，

减少等待的缓冲项的数目，直至相关联延迟测度小于指定临界为止。在解码器侧，必须废除已在编码器侧插入以遵照延迟限制的二进制。对于二进制的此废除而言，可使用基本上与编码器侧的机制相同的机制。

[0384] 在详细描述交错VLC及PIPE编码的位串流的可能性的后，在下文中，描述再次集中于分解成关于图1B、图1C及图2B提及的源符号的以上已提及的语法元素。出于说明的目的，以下描述假定因此而分解的语法元素为绝对变换系数位准。然而，此描述仅仅为实例，且可同样地处理其它类型的语法元素。特定而言，在下文中，描述经由分割且使用基于区块的影像及视讯编码器中的不同熵码的绝对位准的编码。

[0385] 举例而言，通常将视讯序列的图像分解成区块。区块或该区块的彩色分量经由移动补偿预测或者内部预测来预测。区块可具有不同大小且可为方形或者矩形。使用预测参数的相同集合来预测区块的所有取样或区块的彩色分量，该预测参数诸如，参考指针(识别已编码图像集中的参考图像)、移动参数(指定参考图像与当前图像之间的区块的移动的测度)、指定内插滤波器的参数、内部预测模式等。移动参数可经由具有水平及垂直分量的位移向量或经由诸如由6个分量组成的仿射移动参数的较高阶移动参数表示。亦有可能预测参数(诸如，参考指标及移动参数)的多于一个集合与单一区块相关联。在彼状况下，对于预测参数的每一集合而言，产生区块的单一中间预测讯号或区块的彩色分量，且经由中间预测讯号的加权和来构建最终预测讯号。加权参数及亦可能的恒定偏差(向加权和添加该恒定偏差)可固定用于图像或参考图像或参考图像的集合，或者加权参数及亦可能的恒定偏差可包括于相应区块的预测参数的集合中。类似地，亦通常将静止影像分解成区块，且经由内部预测方法(该内部预测方法可为空间内部预测方法或预测区块的DC分量的简单内部预测方法)来预测该区块。在转角情况下，预测讯号亦可为零。

[0386] 通常变换且量化原有区块或原有区块的彩色分量与相应预测讯号之间的差异，该差异亦称为残留讯号。将二维变换应用于残留讯号，且量化所得变换系数。对于此变换编码而言，区块或该区块的彩色分量可在应用变换之前进一步分裂，已将预测参数的特定集合用于该区块或该区块的彩色分量。变换区块可等于或小于用于预测的区块。亦有可能变换区块包括用于预测的区块中的多于一个区块。视讯序列的静止影像或图像中的不同变换区块可具有不同大小，且该变换区块可表示方形或矩形区块。

[0387] 所有该预测及残留参数可分别形成语法元素的串流138及226。

[0388] 可随后使用经由以上编码方案中的任一编码方案的熵编码来传输所得经量化变换系数，该所得经量化变换系数亦称为变换系数位准。为此，可使用扫描将变换系数位准的区块映射至变换系数值的向量(亦即，有序集)上，其中不同扫描可用于不同区块。通常使用锯齿形扫描。对于含有仅经交错讯框的一个字段的取样的区块(该区块可为经编码字段中的区块或经编码讯框中的字段区块)而言，使用针对字段区块具体设计的不同扫描亦十分常见。用于编码变换系数的所得有序序列的可能编码方案为执行位准(run-level)编码。通常，大量变换系数位准为零，且等于零的连续变换系数位准的集合可经由编码等于零(执行)的连续变换系数位准的数目由相应语法元素来有效率地表示。对于剩余(非零的)变换系数而言，以相应语法元素的形式编码实际位准。存在执行位准码的各种替代物。可使用单一语法元素一起编码非零系数之前的执行及非零变换系数的位准。通常，包括区块末端的特殊语法元素，在最后的非零变换系数的后发送该区块末端。或有可能首先编码非零变换

系数位准的数目，且取决于此数目，编码位准及执行。

[0389] 稍微不同的方法用于H.264/AVC中高效的CABAC熵编码中。此处，变换系数位准的编码分成三个步骤。在第一步骤中，针对每一变换区块传输二进制语法元素coded_block_flag，该二进制语法元素用讯号发出该变换区块是否含有重要变换系数位准(亦即，非零的变换系数)。若此语法元素指示存在重要变换系数位准，则编码具有二进制值的影像重要图，具有二进制值的该影像重要图指定哪个变换系数位准具有非零值。且随后按逆向扫描次序编码非零变换系数位准的值。将影像重要图编码成如下语法元素串流138。针对按扫描次序的每一系数，编码二进制语法元素significant_coeff_flag，该二进制语法元素significant_coeff_flag指定相应的变换系数位准是否不等于零。若significant_coeff_flag二进制等于一，亦即，若非零变换系数位准存在于此扫描位置处，则编码另一二进制语法元素last_significant_coeff_flag。此二进制指示当前重要变换系数位准是否为区块内部的最后的重要变换系数位准或另外的重要变换系数位准是否按扫描次序紧接于的后。若last_significant_coeff_flag指示无另外的重要变换系数紧接于的后，则不编码用于指定区块的影像重要图的另外语法元素。在下一个步骤中，编码重要变换系数位准的值，该重要变换系数位准的值在区块内部的位置已经由影像重要图决定。经由使用以下三个语法元素，按逆向扫描次序编码重要变换系数位准的值。二进制语法元素coeff_abs_greater_one指示重要变换系数位准的绝对值是否大于一。若二进制语法元素coeff_abs_greater_one指示该绝对值大于一，则发送另一语法元素coeff_abs_level_minus_two，该另一语法元素coeff_abs_level_minus_two指定变换系数位准减去二的绝对值。此为语法元素的种类，在以下实施例中根据图1B、图1C及图2B执行该语法元素的处理。最终，针对每一重要变换系数位准编码二进制语法元素coeff_sign_flag，该二进制语法元素coeff_sign_flag指定变换系数值的正负号。应再次注意，按扫描次序编码与影像重要图有关的语法元素，而按逆向扫描次序编码与变换系数位准的实际值有关的语法元素，从而允许使用更适合的上下文模型。亦可能将适应性扫描图案用于H.265/HEVC的第一测试模型中的影像重要图。另一概念用于编码H.265/HEVC的第一测试模型中大于4×4的变换区块的绝对变换系数位准。在变换区块大于4×4的情况下，将较大的变换区块分割成4×4区块，且按扫描次序编码该4×4区块，而针对每一4×4区块，使用逆向扫描次序。

[0390] 在H.264/AVC中的CABAC熵编码中，使用二进制机率模型化来编码变换系数位准的所有语法元素。例如，首先二进制化非二进制语法元素coeff_abs_level_minus_two，亦即，将该非二进制语法元素coeff_abs_level_minus_two映像至二元决策(二进制)的序列上，且顺序地编码该二进制。直接编码二进制语法元素significant_coeff_flag、last_significant_coeff_flag、coeff_abs_greater_one及coeff_sign_flag。每一经编码二进制(包括二进制语法元素)与上下文相关联。上下文表示经编码二进制的级别的机率模型。基于使用相应上下文已编码的二进制的值，针对每一上下文估计与两个可能二进制值中的一个二进制值的机率有关的测度。对于与变换编码有关的若干二进制而言，基于已传输语法元素或基于区块内部的位置，选择用于编码的上下文。

[0391] 在编码影像重要图的后，按逆向扫描次序处理区块。如之前所提及，另一概念用于H.265/HEVC的第一测试模型中。将大于4×4的变换区块分割成4×4区块，且按扫描次序处理所得4×4区块，而按逆向扫描次序编码该4×4区块的系数。以下描述对H.265/HEVC及

H.264/AVC的第一测试模型中的所有4×4区块有效且亦对H.264/AVC中8×8区块有效，且此描述亦可分别应用于语法元素的串流138及226的构造。

[0392] 若扫描位置十分重要，亦即，系数不同于零，则在串流138内传输二进制语法元素coeff_abs_greater_one。最初(在区块内)，相应上下文模型组的第二上下文模型选定用于coeff_abs_greater_one语法元素。若区块内部的任何coeff_abs_greater_one语法元素的经编码值等于一(亦即，绝对系数大于2)，则上下文模型化反向切换至集合的第一上下文模型且使用此上下文模型直至区块的末端。否则(区块内部coeff_abs_greater_one的所有经编码值为零且相应的绝对系数位准等于一)，取决于等于零的coeff_abs_greater_one语法元素的数目而选择上下文模型，已在所考虑区块的逆向扫描中处理该coeff_abs_greater_one语法元素。语法元素coeff_abs_greater_one的上下文模型选择可经由以下方程序来概述，其中基于先前的上下文模型标示C_t及先前经编码语法元素coeff_abs_greater_one的值选择当前上下文模型标示C_{t+1}，先前经编码语法元素coeff_abs_greater_one的该值经由方程式中的bin_t表示。针对区块内部的第一语法元素coeff_abs_greater_one，将上下文模型标示设定成等于$C_1=1$。

[0393] $$C_{t+1}(C_t,bin_t)=\begin{cases}0 & bin_t=1\\ \min(C_t+1,4) & bin_t=0\end{cases}$$

[0394] 当相同扫描位置的coeff_abs_greater_one语法元素等于一时，仅编码用于编码绝对变换系数位准的第二语法元素coeff_abs_level_minus_two。将非二进制语法元素coeff_abs_level_minus_two二进制化成二进制的序列，且针对此二进制化的第一二进制，如下文中所描述的选择上下文模型标示。使用固定上下文编码二进制化的剩余二进制。如下选择二进制化的第一二进制的上下文。针对第一coeff_abs_level_minus_two语法元素，选择coeff_abs_level_minus_two语法元素的第一二进制的上下文模型的集合的第一上下文模型，将相应上下文模型标示设定成等于$C_1=0$。针对coeff_abs_level_minus_two语法元素的每一另外的第一二进制，上下文模型化切换至集合中的下一个上下文模型，其中集合中的上下文模型的数目限于5。上下文模型选择可经由以下公式表达，其中基于先前的上下文模型标示C_t选择当前上下文模型标示C_{t+1}。

[0395] $C_{t+1}(C_t)=\min(C_t+1,4)$

[0396] 如之前所提及，针对区块内部的第一语法元素coeff_abs_remain_minus_two，将上下文模型标示设定成等于$C_t=0$。注意针对语法元素coeff_abs_greater_one及coeff_abs_remain_minus_two定义上下文模型湾的不同集合。亦注意，对于H.265/HEVC的第一测试模型而言，可将大于4×4的变换区块分割成4×4区块。可按扫描次序处理分割的4×4区块，且对于每一分割的4×4区块而言，可基于先前的4×4区块中的大于一的系数的数目获得上下文集合。对于大于4×4的变换区块的第一4×4区块而言且对于开始的4×4变换区块而言，可使用单独的上下文集合。

[0397] 亦即，当曾经在以下描述中，基于上下文的编码用于源符号中的任一源符号时，随后可由例如分配器114及212及VLC编码器102/解码器202使用此上下文推导，根据以下实施例coeff_abs_greater_one及coeff_abs_remain_minus_two分解成该源符号。

[0398] 为了降低关于经由CABAC或PIPE处理的二进制的数目与先前技术相比较的复杂性及亦关于计算复杂性或亦为增加编码效率，以下概括的实施例经由使用影像及视讯编码器

及解码器中的不同分区140_1至140_3的不同可变长度码，来描述用于编码绝对位准的方法。然而，以下概括的实施例可应用于影像及视讯编码器的各种绝对位准，如移动向量差异或适应性回路滤波器的系数。尽管如以下概括的执行变换系数位准的编码，但影像重要图的编码可保持如H.265/HEVC的第一测试模型中或如以上所述，或亦可如H.264/AVC中执行影像重要图的编码或相反。

[0399] 如以上所述，在多个分区140_{1-3}中执行绝对变换位准的编码。编码方案示例性地图示于具有三个分区140_{1-3}的图1B中。方案的界限142及144为可变的，从而产生可变的分区大小。如下执行编码方案。

[0400] 第一熵码用以编码第一分量或源符号，亦即，在绝对变换系数位准(z)小于limit1的情况下编码绝对变换系数位准(z)，或在绝对变换系数位准(z)不小于limit1的情况下编码limit1。若绝对变换系数位准大于或等于第一分区140_1的界限limit1，则自绝对变换系数位准减去第一分区(140_1)的界限limit1(142)，且用第二熵码编码所得值z'。若剩余绝对变换系数位准z'大于或等于第二分区140_2的界限limit2-limit1，则再次自绝对变换系数位准z'减去第二分区的界限limit2-limit1，且用第三熵码编码所得值。一般而言，当达到分区的界限时，到达界限的下一个分区的熵码用以编码由绝对变换系数位准减去相应分区的界限产生的值。

[0401] 该熵码可为如执行长度码或查找表的简单可变长度码(例如，霍夫曼码)或采用如CABAC或PIPE的机率模型的更复杂熵码。分区的数目及分区的界限可为可变的或取决于实际语法元素。图1B中所示的分割概念具有以下益处。在下文中，例如，我们使用绝对变换系数，但应理解，该绝对变换系数可用任何其它语法元素来替代。例如，绝对变换系数位准的机率分布可近似为几何分布。因此，经最佳化用于几何分布的熵码可用于编码绝对变换系数位准。即使采用上下文模型化及机率模型选择，但此模型并非始终为局部最佳的。举例而言，尽管归因于大数法则，模型可适用于(具有某一准确度)影像或视讯中特定量的区块，但对于变换区块而言，若该变换区块含有相同量的较低及中间范围变换系数位准，则局部绝对变换系数位准紧接于根本不是几何学的分布的后。针对此情况，几何分布不是适合的模型。亦，当考虑具有较大值的绝对变换系数位准时，该较大值的分布通常是均匀的。分割概念因不同绝对变换系数位准而允许不同机率模型。尽管对于较大的绝对位准而言，不太复杂的熵码可用于降低复杂性，但对于较小的绝对值而言，可应用更复杂的熵码以产生较高效率。

[0402] 如之前所提及，使用适合于不同分区的熵码。在本发明的较佳实施例中，采用三个类型的熵码。第一熵码使用PIPE。然而，应注意，根据替代性实施例，可替代性地使用如CABAC的熵编码方法或任何其它算术编码器。亦即，可经由PIPE编码路径来编码第一符号s_1(参见图2B)。再划分器100及复合器220相应地作用。

[0403] 对于第二类型而言，可使用哥伦布码及包括子集(例如，Golomb-Rice码)的一些截断式变体。亦即，可经由VLC编码器102/解码器202中的该VLC码来编码第二符号s_2(参见图2B)。再划分器100及复合器220相应地作用。

[0404] 指数哥伦布码用作第三类型。亦即，可经由VLC编码器102/解码器202中的该VLC码来编码第三符号s_3(参见图2B)。再划分器100及复合器220相应地作用。不同的VLC码及VLC与PIPE或VLC与算术码的不同组合是可行的。

[0405] 当第一码更复杂但产生更佳的压缩效能时，第二熵码表示复杂性与效能之间的合理取舍。最后的熵码(例如，指数哥伦布码)复杂性较低。在下文中，描述不同分区的编码。

[0406] 若欲使用熵编码器来熵编码诸如分区140_1的分区(诸如，符号s_1)，该熵编码器采用如PIPE编码器104的机率模型(此外，CABAC或任何其它算术编码器可用于此处未进一步描述的替代性实施例中)，则再划分器120将该分区导向PIPE编码器104。首先，可使用二进制化方法在符号化器122中二进制化具有非二进制值的绝对变换系数位准。二进制化将具有非二进制值的绝对变换系数位准映射至二进制的序列。用由分配器114选定的上下文来编码二进制字符串的每一二进制。上下文模型化可针对第一二进制执行且固定用于如H.264/AVC中coeff_abs_level_minus_two的二进制序列的以下二进制，或不同的上下文模型化可用于二进制字符串的每一二进制。注意到二进制化可为如哥伦布码或指数哥伦布码的可变长度码或其它可变长度码。

[0407] 接下来，此处可在VLC编码器102中用哥伦布码来编码诸如分区140_2或符号s_2的分区，且该分区分别在VLC解码器中解码。哥伦布码为针对几何分布源设计的熵码的集合。若哥伦布码的阶数为零，则哥伦布码亦称为一元码。一元码与H.264/AVC中coeff_abs_level_minus_two的二进制化有关。如下建构哥伦布码。针对特定哥伦布参数k，使用整数除法将值n除以哥伦布参数k，且计算余数r。

[0408] $$p=\left\lfloor \frac{n}{k} \right\rfloor$$

[0409] r＝n-pk

[0410] 在导出由以上公式指定的参数的后，可用两个部分编码值n。亦称为前缀部分的第一部分为一元码。所得值p+1指定一的数目及终止零或反的亦然。亦称为余数部分且由r表示的余数值用截断式二进制码来表示。根据特定实施例，Golomb-Rice码用于编码诸如源符号s_2的源符号，该Golomb-Rice码为哥伦布码的子集。亦，当使用含有界限的分区140_{1-3}(诸如，分区140_2)的该熵码时，相应源符号(诸如，源符号s_2)的字母表为有限的，且可修改Golomb-Rice码，以便可改良编码效率。Golomb-Rice码的参数可为固定或可变的。若参数为可变的，则该参数可估计为上下文模型化阶段的部分。举例而言，若源符号s_2输入VLC编码器102，则VLC编码器102可自s_2的上下文决定Golomb-Rice码的参数。Golomb-Rice码为具有二的幂的参数的哥伦布码。因此Golomb-Rice码是基于除以二及乘以二，因此可使用移动及添加操作将该Golomb-Rice码有效率地实施于二进制架构中。Golomb-Rice参数与哥伦布参数之间的关系因此为 $k_{GOLOMB}=2^{k_{RIC}}$。在Golomb-Rice码的情况下，余数部分恰好为余数值的二进制表示。对于零的Golomb-Rice参数而言，所得码与一元码一致且不具有余数部分。对于等于一的参数而言，余数部分由具有两个输入符号的一个二进制组成，该两个输入符号共享相同的一元前缀。在下文中，图示选定的Golomb-Rice参数的一些示例性表。

[0411]

Value	Prefix	Rem.	Prefix	Rem.	Prefix	Rem.	Prefix	Rem.
	k=0		k=1		k=2		k=3	
0	0		0	0	0	00	0	000
1	10		0	1	0	01	0	001
2	110		10	0	0	10	0	010
3	1110		10	1	0	11	0	011
4	11110		110	0	10	00	0	100
5	111110		110	1	10	01	0	101
6	1111110		1110	0	10	10	0	110
7	11111110		1110	1	10	11	0	111
8	111111110		11110	0	110	00	10	000
9	1111111110		11110	1	110	01	10	001
10	11111111110		111110	0	110	10	10	010
11	111111111110		111110	1	110	11	10	011
12	1111111111110		1111110	0	1110	00	10	100
13	11111111111110		1111110	1	1110	01	10	101

[0412] 在分区140_2及Golomb-Rice码的参数的情况下给定分区的范围，诸如，limit2-limit1，可如下执行截断。Golomb-Rice参数描述表示余数部分所需要的二进制的数目，且参数值的二次幂描述可用相同前缀表示的值的数目。该值形成前缀群组。举例而言，对于参数零而言，仅一个前缀可表示一个比值，而对于参数三而言，八个输入值共享相同的前缀，因此前缀群组含有参数三的八个值。对于有限的源字母表及给定的Golomb-Rice码而言，前缀的最后的二进制可针对最后的前缀群组中的值省去，从而产生具有固定长度的前缀码。

[0413] 举例而言，范围可为九，且Golomb-Rice参数为二。针对此示例性情况，可用相同前缀表示的值的数目为四。最大值为九，该最大值亦指示超过界限且必须使用下一个分区的下一个熵码。在此示例性情况下，来自0-3的值具有前缀0，来自4-7的值具有前缀10，且来自8-9的值具有前缀110。因为值8-9形成最后的前缀群组，所以该值8-9的后补零可省去，且该值8-9可经由11表示。换言的，设想源符号s_2输入VLC编码器102，其中s_2的可能值的数目为9(＝limit2-limit1+1)，且彼源符号的Golomb-Rice参数为二。随后，相应Golomb-Rice码字组将经由彼源符号的VLC编码器102输出，该相应Golomb-Rice码字组具有刚刚描述的前缀。针对码字组的余数部分，可如下经由VLC编码器102导出截断码。通常，Golomb-Rice码的参数指示余数部分的二进制的数目。在截断式情况下，并非需要编码余数的所有二进制。针对截断式情况，计数具有固定前缀(例如，省去前缀的二进制)的所有值。注意到，因为码为截断式的，所以计数值始终小于或等于前缀的值的最大数目。若余数部分的截断可行，则最后的前缀群组的截断余数部分的推导可在以下步骤中继续进行。首先，导出小于或等于计数数目的二的幂的最大数目l。随后，在第二步骤中，导出大于计数数目的二的幂的最小数目h。第一值l描述具有h个二进制的余数的前缀群组中值的数目。该值的所有余数以0开始，0的后为余数的二进制表示，该余数的二进制表示限于余数群组中值的数目。针对现视为新前缀群组的最后前缀群组的余数值，执行除所得值形成第一余数群组外的相同步骤，该余数以1开始。执行此步骤，直至导出所有余数为止。例如，范围为14，且参数为三。第一前缀群组含有自0-7的值，且第二前缀群组含有自8-13的值。第二前缀群组含有六个值。参数为l＝2及h＝3。因此，前缀群组的前四个值经由具有三个二进制的余数表示(区分四个值的后补零及二进制表示)。针对最后两个值，再次执行相同步骤。参数为l＝1及h＝2。最后的两个值的余数现可表示为10及11。证明该方法的另一实例为Golomb-Rice参数为四且范围为十。针对此实例，参数为l＝3及h＝4。使用该参数，前八个值的截断余数部分经由四个二进制表

示。剩余两个值具有与之前的先前实例中的余数部分相同的余数部分。若先前实例的范围为九，则第二执行的参数为l＝0及h＝1。剩余的唯一一个值的余数部分为1。

[0414] 第三类型的熵码可为指数哥伦布码。该第三类型的熵码可用于诸如源符号s_3的相等机率分布(例如，具有参数零)。亦即，VLC编码器/解码器对可负责该第三类型的熵码的编码。如之前所提及，较大的绝对变换系数位准通常为均匀分布的。更精确而言，零阶指数哥伦布码可用以编码最后的分区140_3。先前的分区140_2的开始因此界限144可为可变的。界限144的位置可经由依赖于经仔细地编码/解码源符号106、108及/或110或语法元素138(或218、204及/或208或语法元素226)的VLC编码器102/解码器200来控制。

[0415] 在较佳实施例中，如图1B中所示，分区的数目为三，且界限142及144可为可变的。针对第一分区140_1，可如以上论述的采用PIPE编码。然而，或者亦可使用CABAC。在彼状况下，PIPE编码器/解码器对将由二进制算术编码编码器/解码器对来替代。截断式Golomb-Rice码可用于第二分区140_2，且零阶指数哥伦布码可用于最后的分区140_3。

[0416] 在另一较佳实施例中，分区140_{1-3}的数目为三，且第一界限142为固定的，而第二界限144为可变的。针对第一分区140_1，采用CABAC或PIPE。截断式Golomb-Rice码可用于第二分区140_2，且零阶指数哥伦布码可用于最后的分区140_3。

[0417] 在另一较佳实施例中，分区的数目等于二。第一分区140_1可使用CABAC或PIPE。第二分区可使用诸如140_2的Golomb-Rice码。

[0418] 在另一较佳实施例中，分区的数目等于三，而界限142与144两者皆为可变的。例如，针对第一分区140_1，采用CABAC或PIPE，而第二分区140_2可使用截断式Golomb-Rice码，且第三分区140_3使用零阶指数哥伦布码。

[0419] 在较佳实施例中，使用CABAC或PIPE的第一分区140_1的界限142为二，该第一分区140_1采用适应性机率模型。在此较佳实施例中，可如针对以上所述的coeff_abs_greater_one描述的执行第一二进制的上下文模型化，且可如针对以上亦所述的H.264/AVC中的coeff_abs_level_minus_two描述的执行第二二进制的上下文模型化。第二二进制的上下文模型化的上下文决定将分别由分配器114及212来决定。在另一较佳实施例中，使用熵编码的第一分区140_1的界限142为二，该第一分区140_1采用机率模型(例如，PIPE或CABAC)。对于此较佳实施例而言，可如针对以上所述的H.264/AVC中的coeff_abs_greater_one描述的执行第一二进制与第二二进制两者的上下文模型化。可针对第二二进制单独执行针对coeff_abs_greater_one描述的上下文集合评估。

[0420] 在较佳实施例中，使用熵编码的第一分区140_1的界限142可等于一，该第一分区140_1采用机率模型(例如，CABAC或PIPE)。对于相应源符号的二进制字符串的唯一二进制(或字母符号)而言，可如针对先前所述H.264/AVC中的coeff_abs_greater_one描述的执行上下文模型化。

[0421] 在较佳实施例中，使用熵编码的第一分区140_1的界限142可为三，该第一分区140_1采用机率模型(例如，CABAC或PIPE)。可如H.264/AVC中的coeff_abs_greater_one执行相应源符号的二进制字符串的第一及第二二进制的上下文模型化。可如H.264/AVC中的coeff_abs_level_minus_two执行第三二进制的上下文模型化。可针对第二二进制单独执行针对coeff_abs_greater_one描述的上下文集合评估。

[0422] 在较佳实施例中，截断式Golomb-Rice码的集合可用作第二分区140_2的熵码。取决

于熵码参数来指定第三分区140_3的开始的第二分区的界限144可为可变的。亦在此较佳实施例中，Golomb-Rice参数可限制在三，且可如H.264/AVC中的coeff_abs_level_minus_two的上下文模型化执行参数选择。范围limit-limit2可为可变的且可取决于Golomb-Rice参数。若参数为零，则范围为8。对于参数一而言，范围为10。在参数二的情况下，范围为12，且对于参数三而言，范围等于16。在此较佳实施例中，在变换系数的区块开始处将Golomb-Rice参数设定成零。对于大于或等于第一界限的区块中的每一经编码变换系数位准而言，使用相应的Golomb-Rice码。在编码(或解码)位准的后，进行以下评估来更新用于编码(或解码)大于或等于第一界限的下一个位准的Golomb-Rice参数。注意到经由使用调适的此形式无法减小Golomb-Rice参数。

[0423] 可如下概述参数调适规则，其中k_{t+1}表示欲用于编码下一个位准值的Golomb-Rice参数，且$value_t$表示具有相应Golomb-Rice参数k_t的先前经编码的值。

[0424] $$k_{t+1}=\begin{cases}0 & value_t\in[0,1]\wedge k_t<1\\ 1 & value_t\in[2,3]\wedge k_t<2\\ 2 & value_t\in[4,5]\wedge k_t<3\\ 3 & value_t>5\wedge k_t<4\\ k_t & otherwise\end{cases}\quad(QQ)$$

[0425] 在较佳实施例中，截断式Golomb-Rice码的集合可用作第二分区140_2的熵码。取决于熵码参数来指定第三分区140_3的开始的第二分区140_2的界限144可为可变的。亦在此较佳实施例中，Golomb-Rice参数可限制在三，且可如H.264/AVC中的coeff_abs_level_minus_two的上下文模型化执行参数选择。范围可为可变的且取决于Golomb-Rice参数。若参数为零，则范围为8。对于参数一而言，范围为10。在参数二的情况下，范围为12，且对于参数三而言，范围等于16。在此较佳实施例中，在区块开始处将Golomb-Rice参数设定成零。如方程式(QQ)所述执行Golomb-Rice参数调适。注意到经由使用调适的此形式无法减小参数。

[0426] 在另一较佳实施例中，截断式Golomb-Rice码的集合可用作第二分区140_2的熵码。取决于熵码参数来指定第三分区140_3的开始的第二分区140_2的界限144可为固定的。亦在此较佳实施例中，Golomb-Rice参数可限于三，且可如H.264/AVC中的coeff_abs_level_minus_two的上下文模型化执行参数选择。第二分区140_2的范围可固定于14。在此较佳实施例中，可在区块开始处将Golomb-Rice参数设定成零。如方程式(QQ)所述执行Golomb-Rice参数调适。注意到经由使用调适的此形式无法减小参数。

[0427] 在另一较佳实施例中，截断式Golomb-Rice码的集合可用作第二分区140_2的熵码。取决于熵码参数来指定第三分区140_3的开始的第二分区140_2的界限144可为可变的。亦在此较佳实施例中，Golomb-Rice参数可限于三，且可如H.264/AVC中的coeff_abs_level_minus_two的上下文模型化执行参数选择。范围可为可变的且取决于Golomb-Rice参数。若参数可为零，则范围可为8。对于参数一而言，范围可为10。在参数二的情况下，范围可为12，且对于参数三而言，范围可等于16。在此较佳实施例中，可在区块开始处将Golomb-Rice参数设定成零。如方程式(QQ)所述执行Golomb-Rice参数调适。注意到经由使用调适的此形式无法减小参数。且亦注意到直接切换(例如，自零至三)是可行的。在此较佳实施例中，采用机率模型用熵码来编码Golomb-Rice码的前缀部分。可关于H.264/AVC中的coeff_abs_level_minus_two执行上下文模型化。

[0428] 在另一较佳实施例中，固定的Golomb-Rice参数可用以编码当前变换区块中的所有变换系数位准。在此实施例中，可计算先前区块的最佳参数，且可将先前区块的最佳参数用于当前变换区块。对于此实施例而言，范围可固定在14。

[0429] 在另一较佳实施例中，固定的Golomb-Rice参数可用以编码当前变换区块中的所有变换系数位准。在此实施例中，可计算先前区块的最佳参数，且可将先前区块的最佳参数用于当前变换区块。对于此实施例而言，如之前描述的，范围可为可变的。

[0430] 在另一较佳实施例中，若当前扫描标示的已编码(或解码)邻域含有大于先前界限的绝对变换系数位准，则评估该已编码(或解码)邻域。对于此较佳实施例而言，可经由使用局部因果模板中的邻域来导出最佳参数。

[0431] 因此，以上提及的实施例特定描述熵编码设备，该熵编码设备包含：分解器136，该分解器136经配置以经由将语法元素的至少一个子组个别地分解成相应数目n的源符号s_i，来将语法元素的序列138转换成源符号106的序列106，其中i＝1…n，源符号的相应数目n取决于关于将相应语法元素的值范围再划分成n个分区140_{1-3}的序列的哪个分区、相应语法元素的值z属于n个分区140_{1-3}的序列的哪个分区，以便相应数目的源符号s_i的值的和产生z，且若n>1，则对于所有i＝1…n-1而言，s_i的值对应于第i个分区的范围；再划分器100，该再划分器100经配置以将源符号的序列106再划分成源符号的第一子序列108及源符号的第二子序列110，以使得在第一子序列108内含有所有源符号s_x，其中x为{1…n}的第一子集的成员，且在第二子序列110内含有所有源符号s_y，其中y为与第一子集互斥的{1…n}的第二子集的成员；VLC编码器102，该VLC编码器102经配置以用符号的方式编码第一子序列108的源符号；以及PIPE或算术编码器104，该PIPE或算术编码器104经配置以编码源符号的第二子序列110。

[0432] 语法元素的子组的值z可为绝对值。第二子集可为{1}，其中针对所有p,q∈{1..n}其中p>q，n可为3，n个分区的序列经排列以使得第p个分区比第q个分区覆盖值范围中的较高值。第一子集可为{2,3}，其中VLC编码器(102)经配置以使用Golomb-Rice码来用符号的方式编码源符号s_2且使用指数哥伦布码来用符号的方式编码源符号s_3。更大体而言，2可为第一子集的元素，其中VLC编码器(102)经配置以使用Golomb-Rice码来用符号的方式编码源符号s_2且经配置以根据先前经编码源符号调适Golomb-Rice码的Golomb-Rice参数，亦即，k。分解器可经配置以根据先前经编码源符号调适分区之间的一或更多极限。可组合两种调适。亦即，限制第二分区的极限位置可经调适以使得该位置被彼此间隔开，以使得Golomb-Rice码的长度，亦即，Golomb-Rice码的码字组的数目对应于(或规定)第二分区的宽度的长度。在k的调适可分别经由Golomb-Rice码的长度及第二分区的宽度定义分隔第二分区与第三分区的极限的位置的情况下，可以另外方式定义分隔第一分区与第二分区之间的极限，诸如，经由定义固定或根据其它上下文相依性调适。经由耦合k的调适，以便第二分区的宽度对应于Golomb-Rice码的长度，来最佳地使用编码效率。由于诸如指数哥伦布码的复杂度更低的码可用于第三分区，故使k适应于语法元素的统计学能够调适第二分区的宽度，以使得第三分区可覆盖尽可能多，以降低总的编码复杂性。此外，第一分区的长度可限于j∈{1,2,3}可能的语法元素值，诸如，最低的三个位准。可区别编码在考虑之中的语法元素，或在考虑之中的语法元素可表示预测残留，表示预测残留的以上例示的变换系数位准的情况亦是如此。第一源符号s1可经由使用截断式一元码符号化/去符号化，其中所得j个

二进制(该j个二进制的部分或全部)经上下文适应性编码或并非如以上所提及的编码。

[0433] 语法元素的子组可涵盖图像的变换区块的绝对变换系数的绝对变换系数位准,其中相应变换区块的绝对变换系数位准根据引导穿越相应变换区块的绝对变换系数的扫描路径排列于语法元素的序列(138)内,其中分解器可经配置以取决于扫描次序居先的相应变换区块的绝对变换系数的已编码绝对变换系数位准,或取决于当前欲按扫描次序分解的绝对变换系数位准的位置,或基于与当前欲分解的绝对变换系数位准的位置在空间上或者按扫描次序相邻的变换系数的已重建绝对变换系数位准的评估,在分解相应变换区块的绝对变换系数的绝对变换系数位准期间调适分区之间的一或更多极限。

[0434] 此外,以上提及的实施例特定描述熵解码设备,该熵解码设备包含:VLC解码器200,该VLC解码器200经配置以用码字组的方式自第一位串流206的码字组重建源符号第一子序列204的源符号;PIPE或算术解码器202,该PIPE或算术解码器202经配置以重建源符号的第二子序列208;组合器224,该组合器224经配置以经由自相应数目的源符号个别地组合每一语法元素,而自源符号的第一子序列204及源符号的第二子序列208组合语法元素的序列226,其中该组合器经配置以针对语法元素的至少一个子组,决定源符号s_i的相应数目n,其中i=1…n取决于关于将相应语法元素的值范围再划分成n个分区140_{1-3}的序列中的哪个分区、相应语法元素的值z属于n个分区140_{1-3}的序列中的哪个分区,而此决定在只要s_i的值对应于第i个分区的范围时,则经由自1至n合计相应数目的源符号s_i的值,以获取语法元素z的值来进行;其中组合器224经配置以自第一子序列(204)检索所有源符号s_x,其中x为{1…n}的第一子集的成员,且组合器224经配置以自第二子序列208检索所有源符号s_y,其中y为与第一子集互斥的{1…n}的第二子集的成员。语法元素的子组的值z可为绝对值。第二子集可为{1},其中针对所有p,q∈{1..n}其中p>q,n可为3,n个分区的序列经排列以使得第p个分区比第q个分区覆盖值范围中的较高值。第一子集可为{2,3},其中VLC解码器200经配置以使用Golomb-Rice码来用码字组的方式重建源符号s_2且使用指数哥伦布码来用码字组的方式重建源符号s_3。更大体而言,2可为第一子集的元素,其中VLC解码器102经配置以使用Golomb-Rice码来用码字组的方式重建源符号s_2且经配置以根据先前经重建源符号调适Golomb-Rice码的Golomb-Rice参数。熵解码设备可进一步包含复合器220,该复合器220经配置以再结合源符号的第一子序列204及源符号的第二子序列,来获取源符号的序列218。语法元素可为不同类型,且组合器可经配置以取决于语法元素的类型而执行个别组合。语法元素的子组可涵盖图像的变换区块的绝对变换系数的绝对变换系数位准,其中相应变换区块的绝对变换系数位准根据引导穿越相应变换区块的绝对变换系数的扫描路径排列于语法元素的序列138内,其中组合器可经配置以取决于扫描次序居先的相应变换区块的绝对变换系数的已重建绝对变换系数位准,或取决于当前欲按扫描次序组合的绝对变换系数位准的位置,或基于与当前欲组合的绝对变换系数位准的位置在空间上或者按扫描次序相邻的变换系数的已重建绝对变换系数位准的评估,在位准相应变换区块的绝对变换系数的绝对变换系数位准期间调适分区之间的一或更多极限。

[0435] 关于使用根据图1B的分解将PIPE编码与VLC编码组合的实施例,批注下文,以换言的重复该实施例的一些态样。

[0436] 已描述符号的序列映像至位串流且描述逆映射。每一符号载有在编码器及解码器处同时已知的一或更多相关联参数。熵编解码器含有多个先进先出(FIFO)缓冲器,其中将

该多个FIFO缓冲器中的每一FIFO缓冲器分配予与符号相关联的一或更多参数的子集。针对符号的给定的一或更多参数，编码器将符号分配予相应的FIFO缓冲器。由于编码器分配规则在解码器侧已知，故解码器自FIFO缓冲器读取编码器分配予该FIFO缓冲器的符号。

[0437] 使用标准可变长度码来编码一些语法元素，且将该语法元素写入至特定缓冲器。使用机率区间分割熵(PIPE)编码概念来编码其它语法元素。此举的后，首先将符号二进制化，且基于相关联机率估计将所得二进制分类。机率估计可由可在编码器及解码器处同时执行的测度给定或获得。特定FIFO缓冲器含有具有经估计机率值的符号，该经估计机率值属于选定的机率的子集，以便可改良熵编码。经由将PIPE概念与VLC组合来实现的改良为复杂性降低，同时仍提供较高的编码效率。用简单且低复杂性的VLC方法来编码标准VLC码所适合的符号，而用更复杂的PIPE概念来编码其它符号，经由用VLC码编码该其它符号，该其它符号的位率将显著增加。

[0438] 因此，为了进一步降低熵编码的复杂性，将符号分成两个类别。第一类别的符号可较佳地用VLC码表示且不需要更复杂的PIPE编码，而第二类别的符号无法有效率地用VLC码表示，且该符号的PIPE编码显著地降低所需要的位率。

[0439] 应注意，虽然仅用一个VLC编码器及解码器描述以上实施例，但显然此概念可一般化至使用用PIPE编码器交错的两个及两个以上VLC编码器及解码器。

[0440] 尽管已在设备的上下文中描述一些态样，但显然该态样亦表示相应方法的描述，其中区块或装置对应于方法步骤或方法步骤的特征结构。类似地，在方法步骤的上下文中描述的态样亦表示相应区块或项目或相应设备的特征结构的描述。该方法步骤中的一些或全部可经由(或使用)硬设备来执行，硬设备例如，微处理器、可程序化计算机或电子电路。在一些实施例中，最重要方法步骤中的一些一或更多方法步骤可经由此设备执行。

[0441] 本发明的经编码/经压缩讯号可在数字储存媒体上储存或可在诸如无线传输媒体或有线传输媒体的传输媒体上传输，该有线传输媒体诸如因特网。

[0442] 取决于某些实施要求，可在硬件中或在软件中实施本发明的实施例。可使用数字储存媒体(例如，软盘、DVD、蓝光光盘、CD、ROM、PROM、EPROM、EEPROM或闪存)执行实施，该数字储存媒体上储存有电子可读取的控制讯号，该电子可读取的控制讯号与可程序化计算机系统合作(或能够合作)，以使得执行相应方法。因此，数字储存媒体可为计算机可读取的。

[0443] 根据本发明的一些实施例包含数据载体，该数据载体具有电子可读取控制讯号，该电子可读取控制讯号能够与可程序化计算机系统合作，以使得执行本文所述的方法中的一种方法。

[0444] 大体而言，本发明的实施例可实施为具有程序代码的计算机程序产品，当计算机程序产品在计算机上执行时，该程序代码经操作用于执行该方法中的一种方法。例如，程序代码可在机器可读取载体上储存。

[0445] 其它实施例包含在机器可读取载体上储存的用于执行本文所述方法中的一种方法的计算机程序。

[0446] 换言的，本发明方法的实施例因此为计算机程序，该计算机程序具有用于当计算机程序在计算机上执行时，执行本文所述的方法中的一种方法的程序代码。

[0447] 本发明方法的另一实施例因此为数据载体(或数字储存媒体或计算机可读取媒体)，该数据载体包含该数据载体上记录的用于执行本文所述方法中的一种方法的计算机

程序。

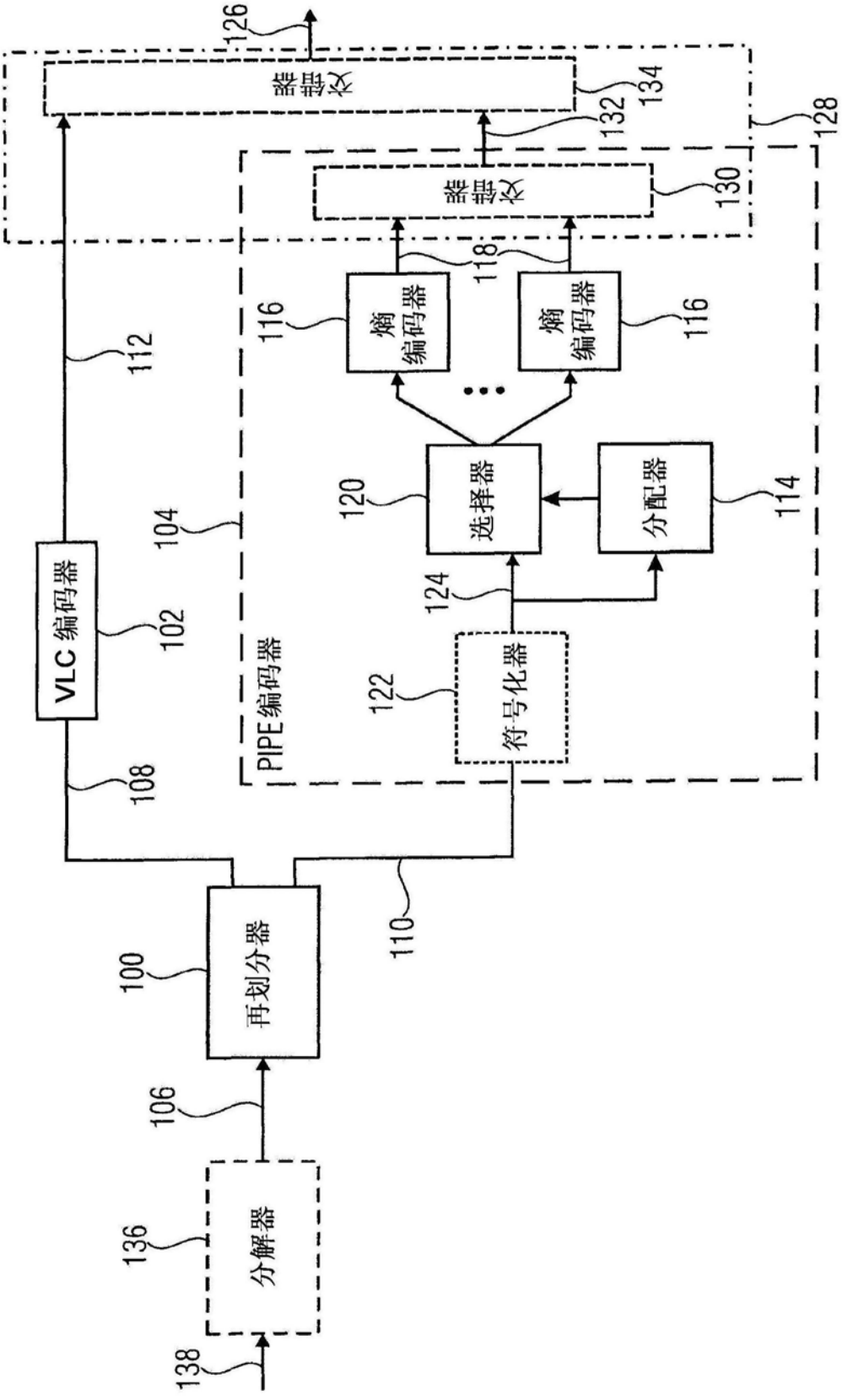
[0448] 本发明方法的另一实施例因此为表示用于执行本文所述方法中的一种方法的计算机程序的数据串流或讯号的序列。例如，数据串流或讯号的序列可经配置以经由数据通讯连接转移，例如经由因特网。

[0449] 另一实施例包含处理手段，例如，计算机或可程序逻辑装置，该处理手段经配置以或经调适以执行本文所述方法中的一种方法。

[0450] 另一实施例包含计算机，该计算机上安装有用于执行本文所述方法中的一种方法的计算机程序。

[0451] 在一些实施例中，可程序逻辑装置(例如，现场可程序门阵列)可用以执行本文所述方法的功能中的一些或全部。在一些实施例中，现场可程序门阵列可与微处理器合作，以执行本文所述方法中的一种方法。大体而言，方法较佳地经由任何硬设备来执行。

[0452] 以上描述的实施例仅说明本发明的原理。应理解，本文所述的布置及细节的修改及变化将对其他熟习此项技术者显而易见。因此，本发明仅受到以下即将描述的专利申请范围的范畴的限制而不受以本文实施例的描述及阐释的方式呈现的具体细节的限制。



126
交错器
134
132
128
交错器
130
118
116
熵编码器
熵编码器
116
112
104
120
选择器
分配器
114
VLC 编码器
102
124
PIPE 编码器
122
符号化器
108
110
100
再划分器
106
136
分解器
138

图1A

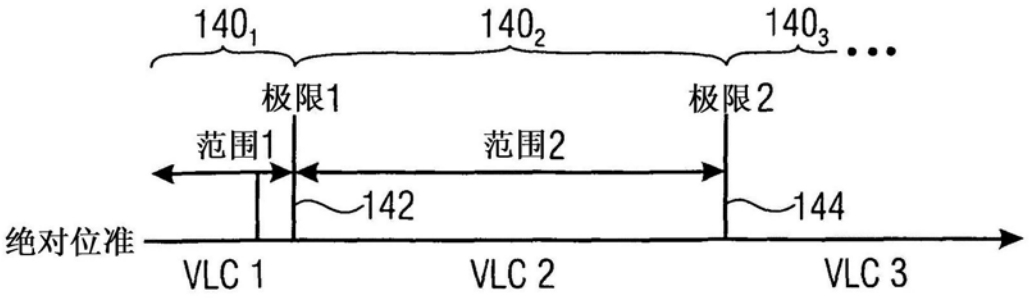

140₁
140₂
140₃
极限1
极限2
范围1
范围2
142
144
绝对位准
VLC 1
VLC 2
VLC 3

图1B

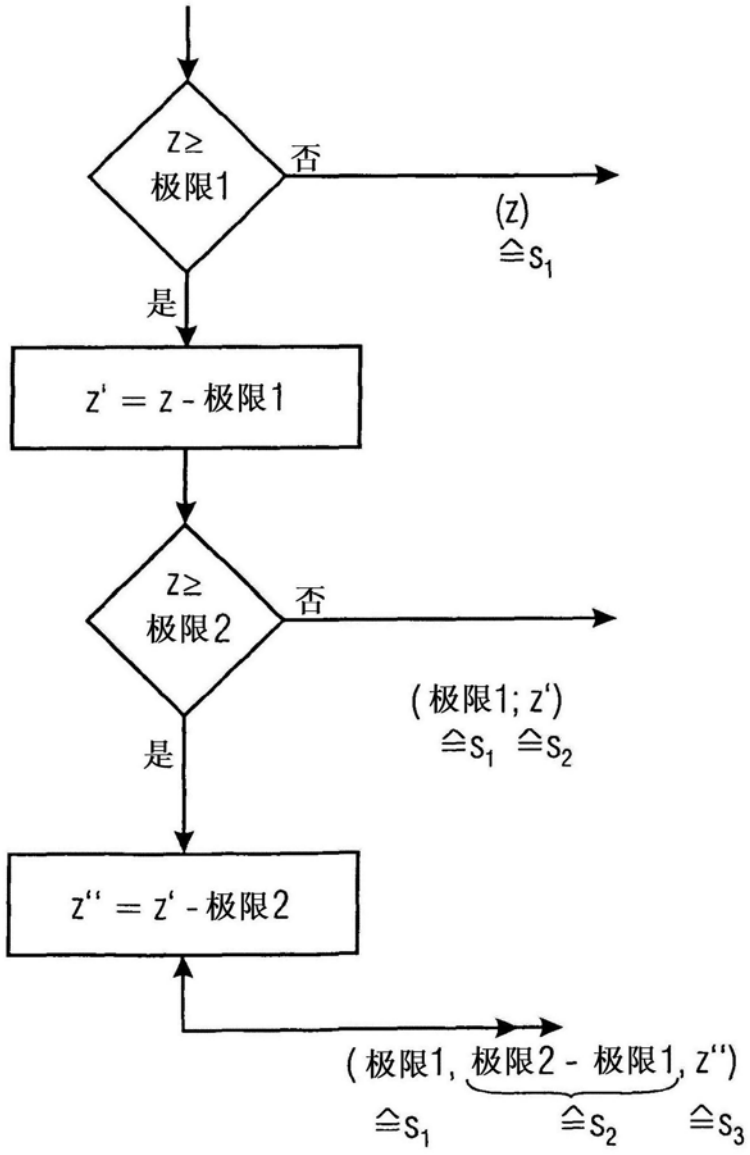

z≥
极限1
否
(z)
≙S1
是
z' = z - 极限1
z≥
极限2
否
(极限1; z')
≙S1 ≙S2
是
z'' = z' - 极限2
(极限1, 极限2 - 极限1, z'')
≙S1 ≙S2 ≙S3

图1C

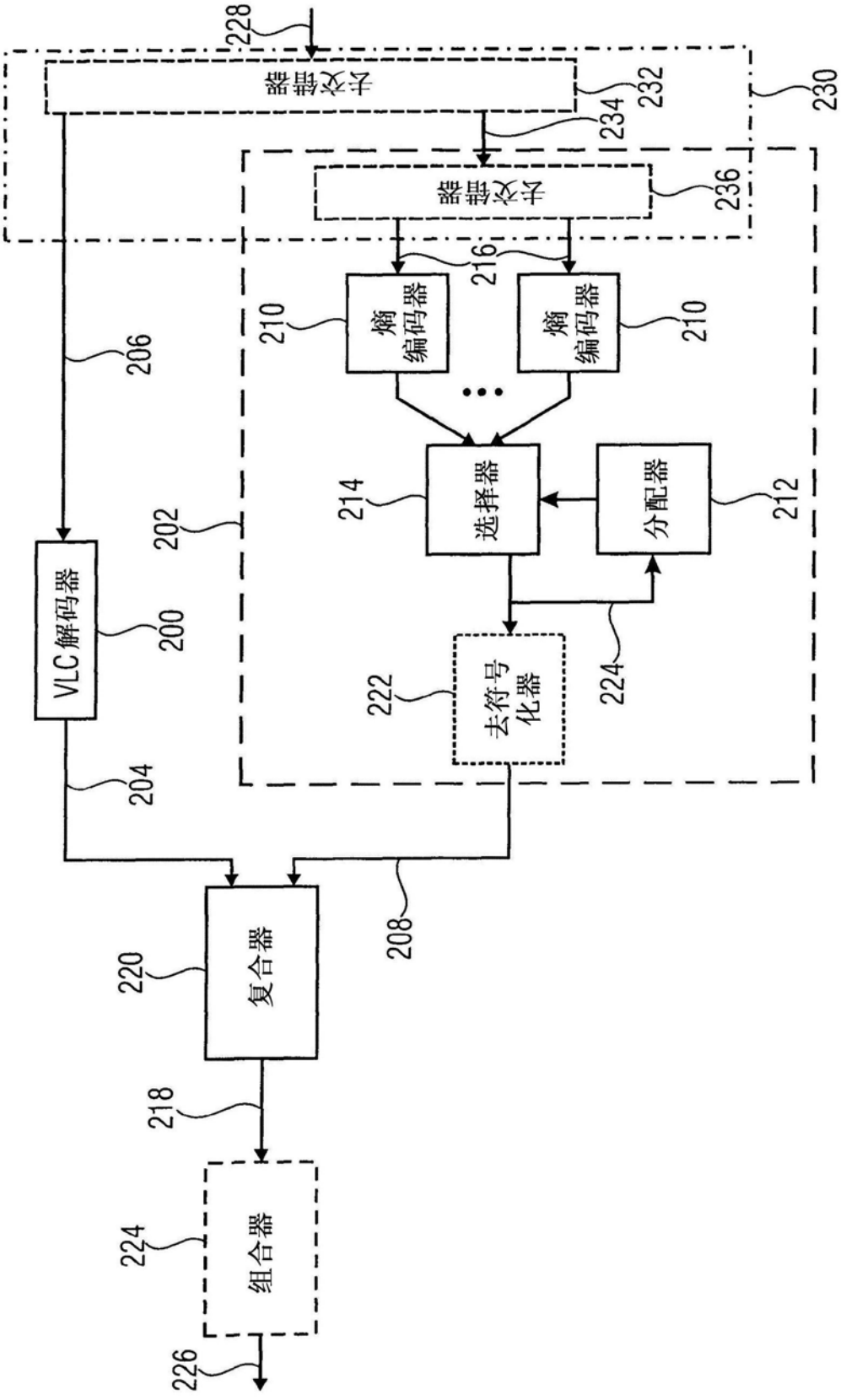

228
去交错器
232
234
230
去交错器
236
216
熵编码器
210
熵编码器
210
206
202
选择器
214
分配器
212
VLC 解码器
200
222
去符号化器
224
204
208
复合器
220
218
组合器
224
226

图2A

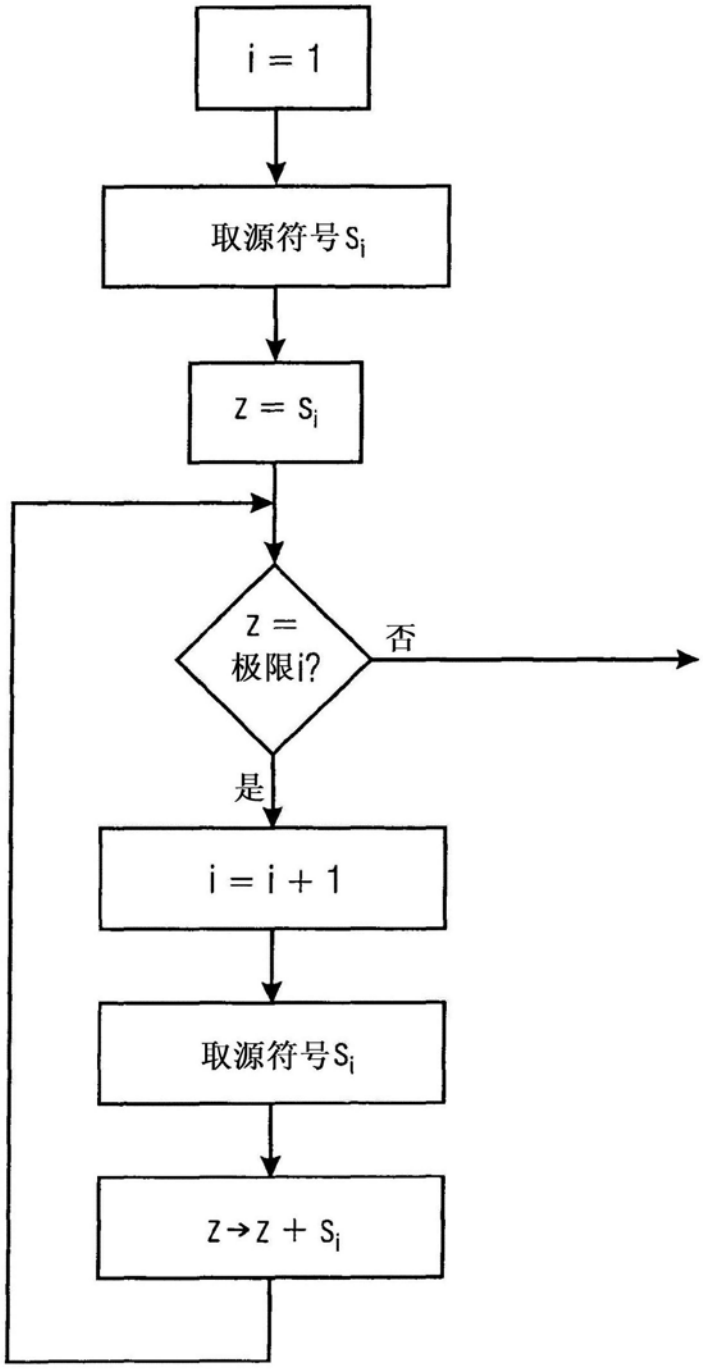


图2B

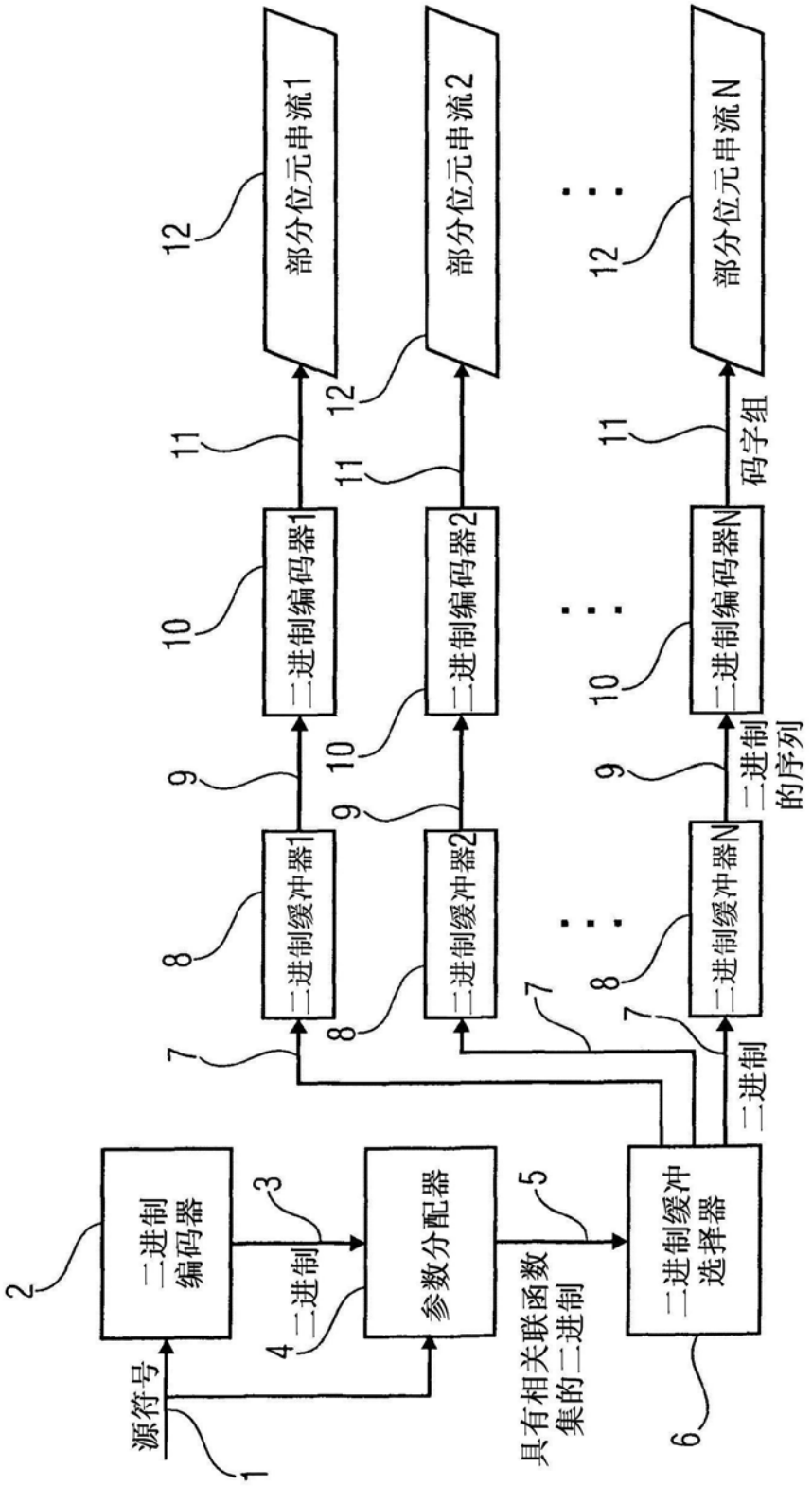

源符号
1
二进制编码器
2
二进制
3
二进制
4
参数分配器
5
具有相关联函数集的二进制
二进制缓冲选择器
6
7
二进制缓冲器1
8
二进制编码器1
9
10
部分位元串流1
11
12
二进制缓冲器2
二进制编码器2
部分位元串流2
二进制缓冲器N
二进制编码器N
部分位元串流N
二进制的序列
码字组

图3

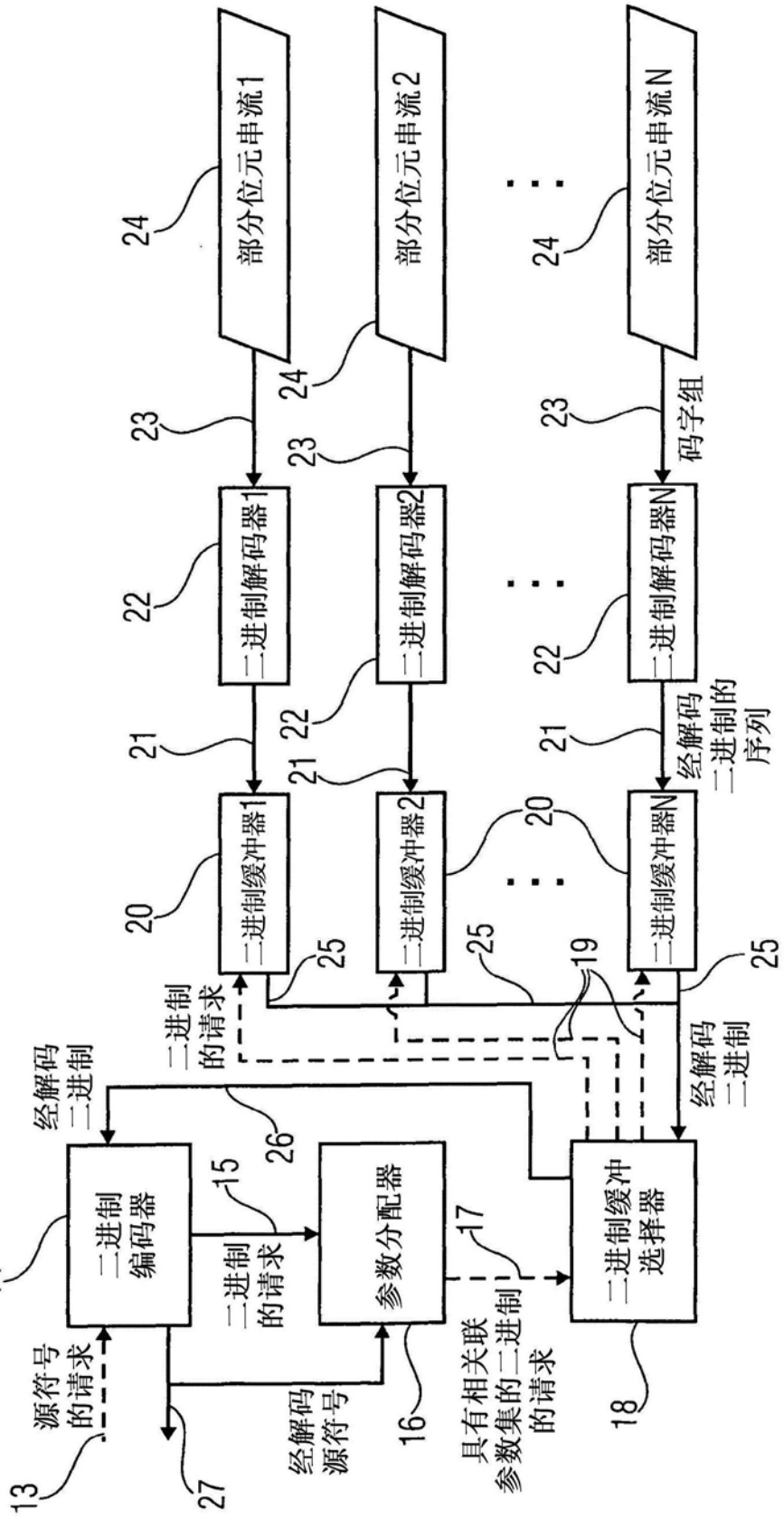

部分位元串流1
部分位元串流2
部分位元串流N
码字组
二进制解码器1
二进制解码器2
二进制解码器N
经解码二进制的序列
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
二进制的请求
经解码二进制
二进制编码器
二进制的请求
参数分配器
具有相关联参数集的二进制的请求
二进制缓冲选择器
源符号的请求
经解码源符号
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27

图4

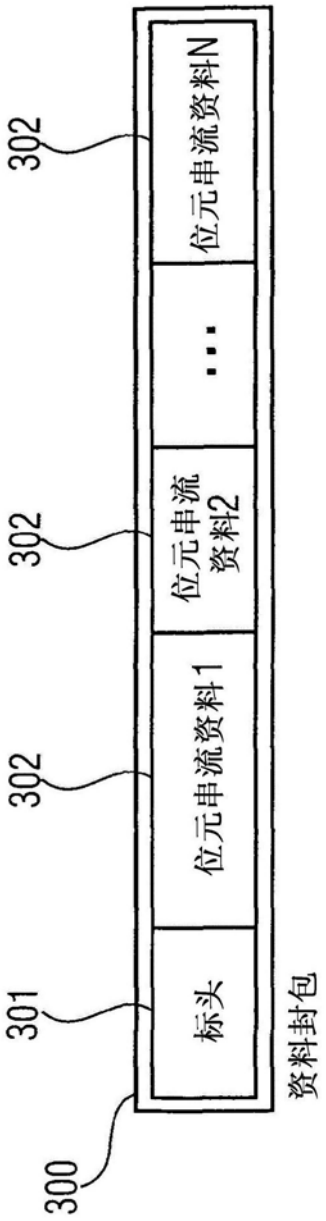


图5

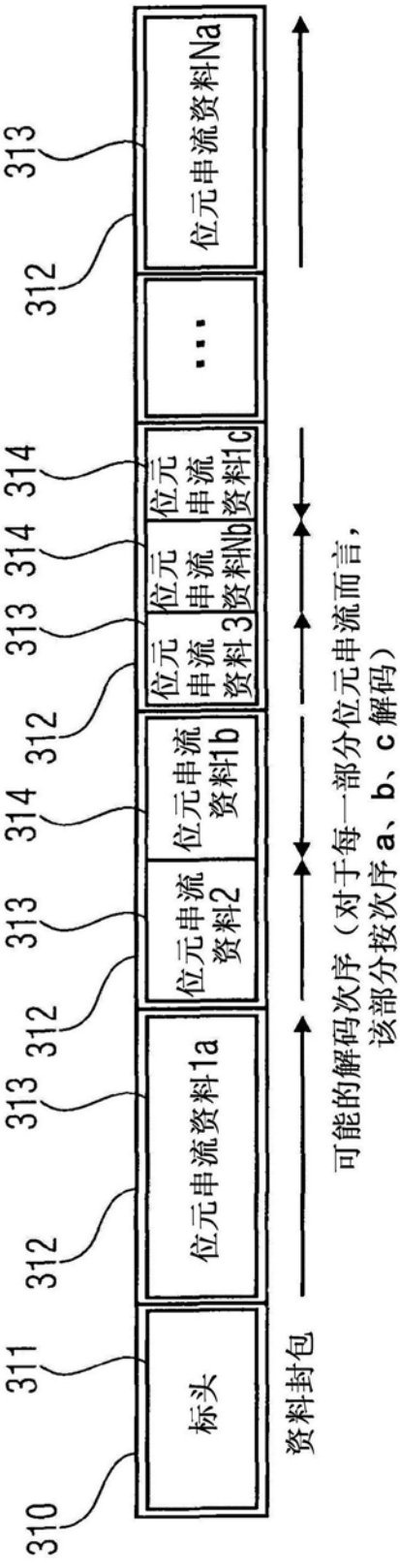

310
311
312
313
314
标头
位元串流资料1a
位元串流资料2
位元串流资料1b
位元串流资料3
位元串流资料Nb
位元串流资料1c
...
位元串流资料Na
资料封包
可能的解码次序（对于每一部分位元串流而言，该部分按次序 a、b、c 解码）

图6

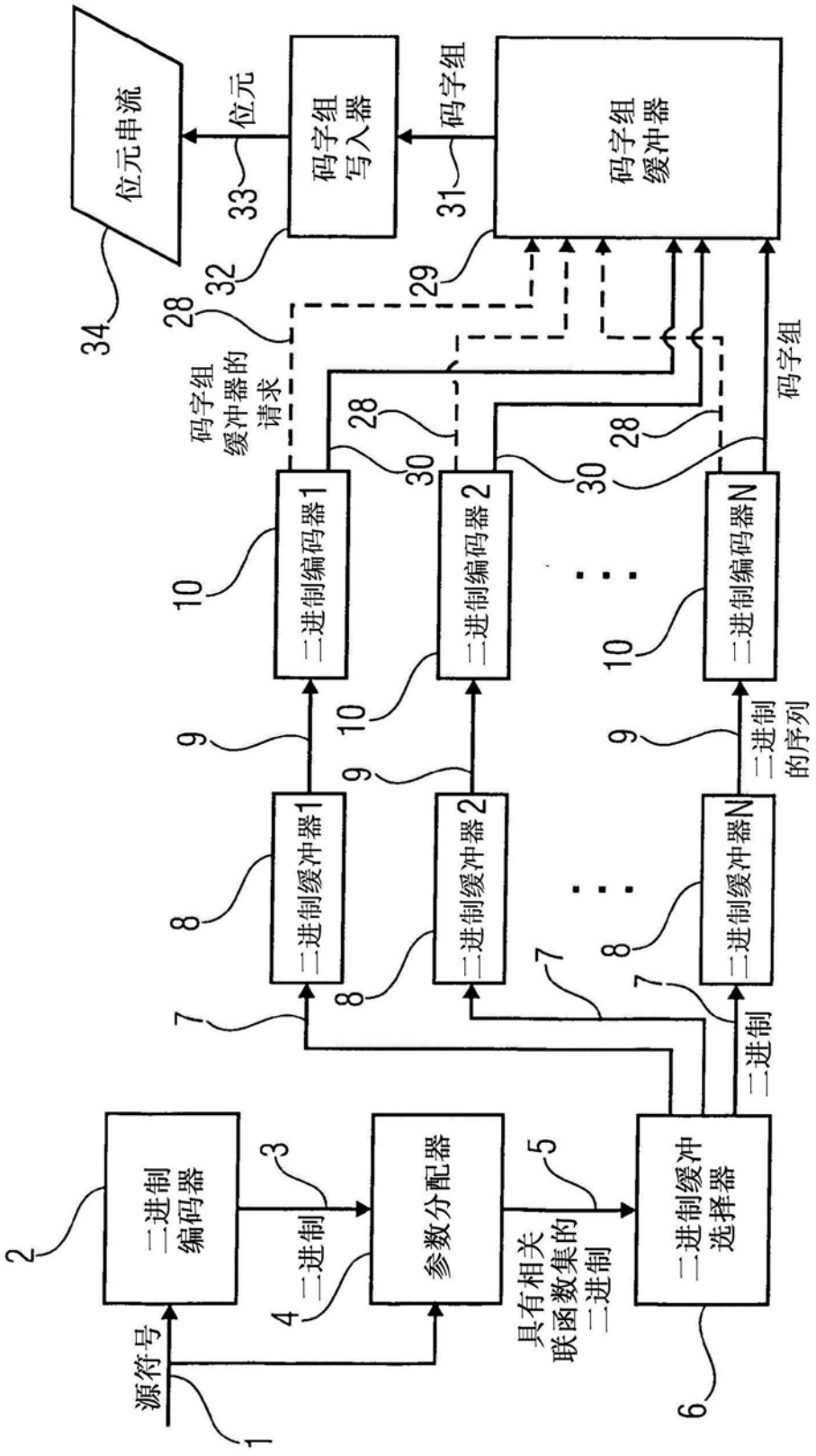

源符号
1
二进制编码器
2
二进制
3
二进制
4
参数分配器
5
具有相关联函数集的二进制
二进制缓冲选择器
6
二进制
7
二进制缓冲器1
8
二进制缓冲器2
二进制缓冲器N
9
二进制编码器1
10
二进制编码器2
二进制编码器N
二进制的序列
码字组缓冲器的请求
28
30
码字组
29
码字组缓冲器
31
码字组写入器
32
位元
33
位元串流
34

图7

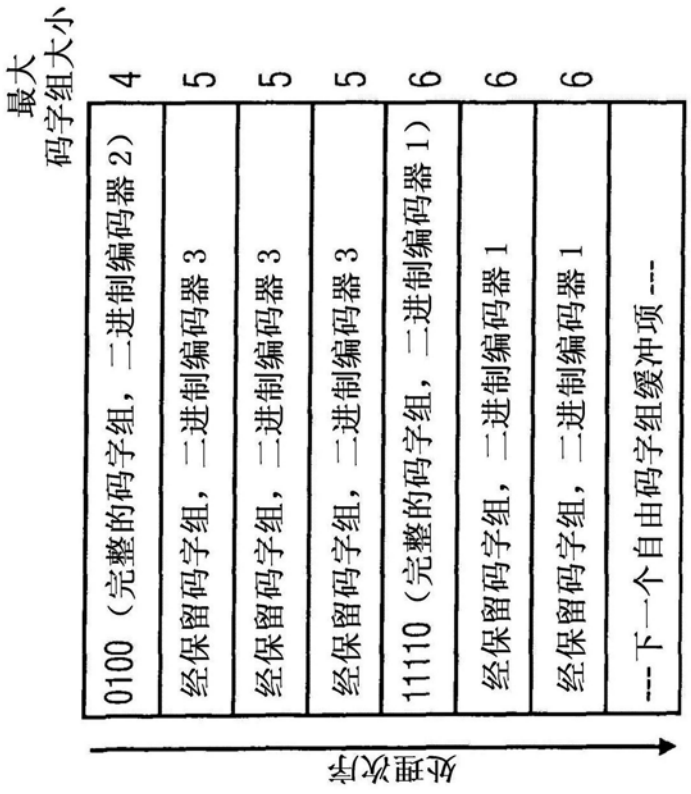

最大
码字组大小
0100（完整的码字组，二进制编码器 2） 4
经保留码字组，二进制编码器 3 5
经保留码字组，二进制编码器 3 5
经保留码字组，二进制编码器 3 5
11110（完整的码字组，二进制编码器 1） 6
经保留码字组，二进制编码器 1 6
经保留码字组，二进制编码器 1 6
---下一个自由码字组缓冲项 ---
处理次序

图8A

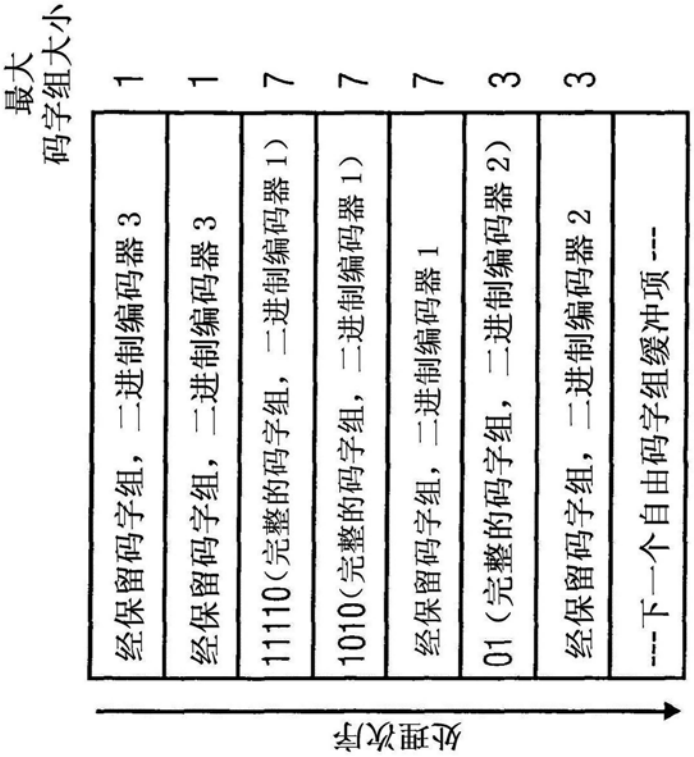

最大
码字组大小
经保留码字组，二进制编码器 3 1
经保留码字组，二进制编码器 3 1
11110（完整的码字组，二进制编码器 1） 7
1010（完整的码字组，二进制编码器 1） 7
经保留码字组，二进制编码器 1 7
01（完整的码字组，二进制编码器 2） 3
经保留码字组，二进制编码器 2 3
---下一个自由码字组缓冲项 ---
处理次序

图8B

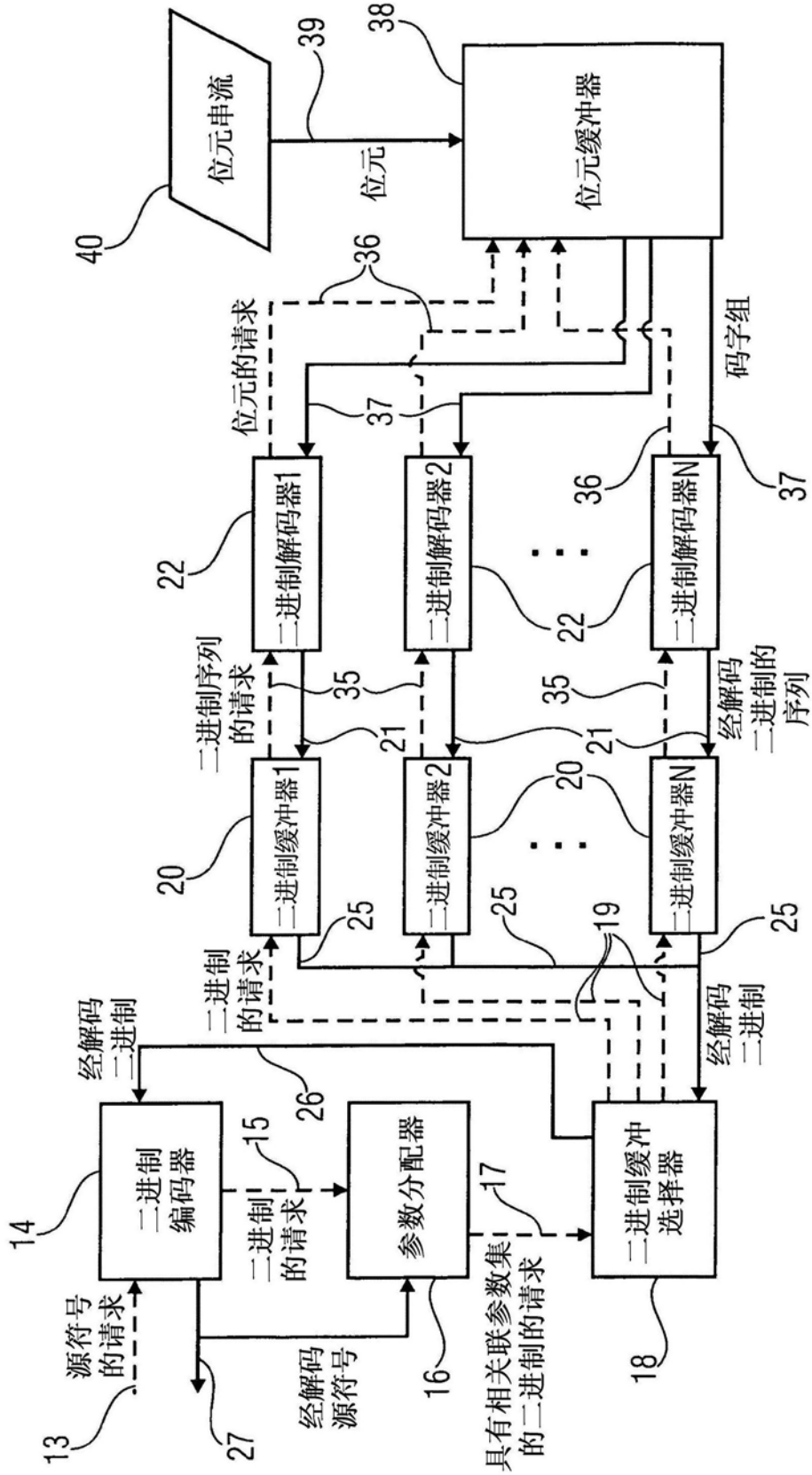

位元串流
位元
位元缓冲器
二进制解码器1
二进制解码器2
二进制解码器N
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
二进制编码器
参数分配器
二进制缓冲选择器
位元的请求
码字组
二进制序列的请求
经解码二进制的序列
二进制的请求
经解码二进制
二进制的请求
经解码二进制
源符号的请求
经解码源符号
具有相关联参数集的二进制的请求
13
14
15
16
17
18
19
20
21
22
25
26
27
35
36
37
38
39
40

图9

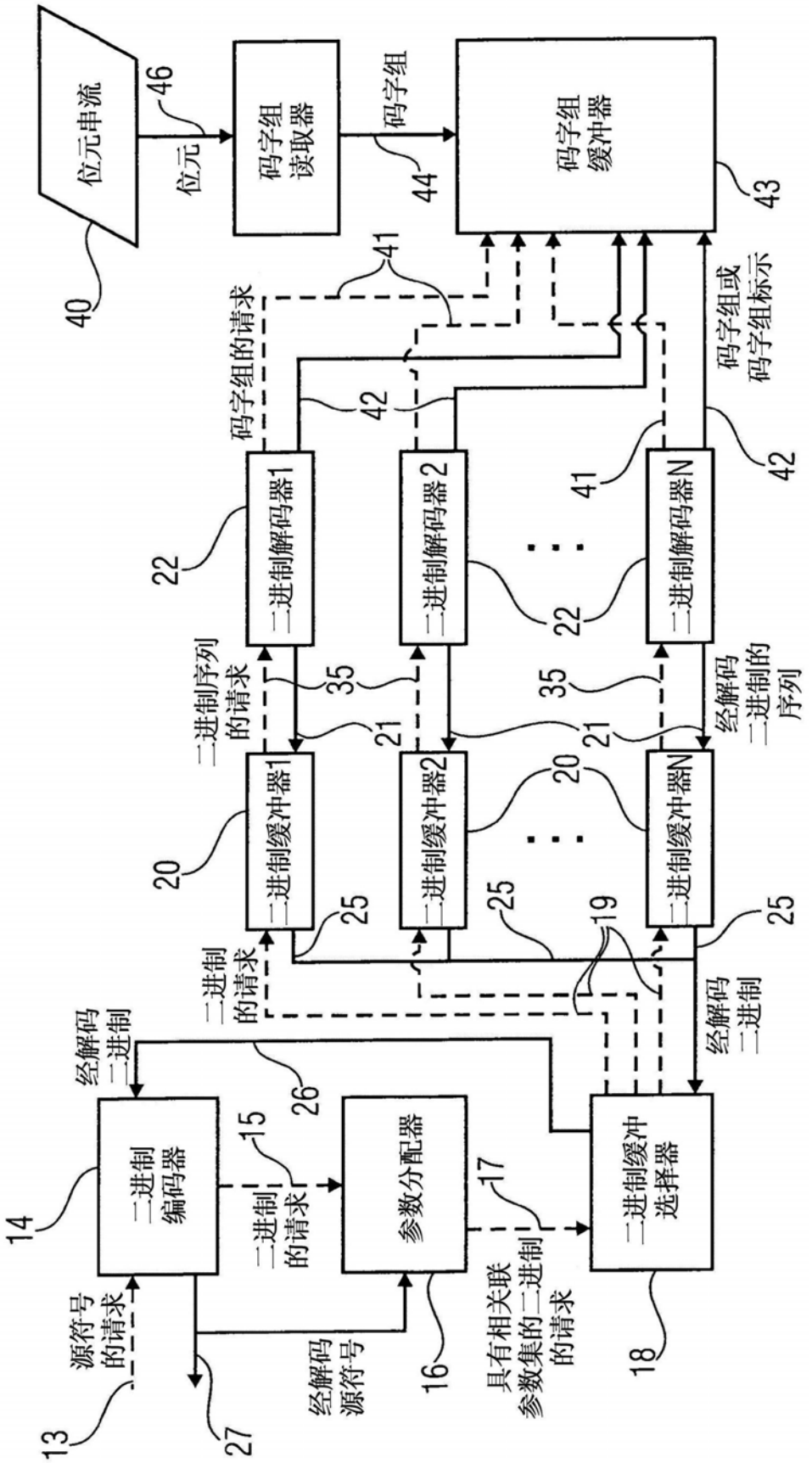

位元串流
40
46
位元
码字组
读取器
44
码字组
缓冲器
43
码字组的请求
41
42
码字组或
码字组标示
二进制解码器1
二进制解码器2
二进制解码器N
22
二进制序列
的请求
35
21
经解码
二进制的
序列
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
20
25
19
二进制
的请求
经解码
二进制
二进制
编码器
14
26
15
二进制
的请求
参数分配器
17
二进制缓冲
选择器
18
源符号
的请求
13
27
经解码
源符号
16
具有相关联
参数集的二进制
的请求

图10

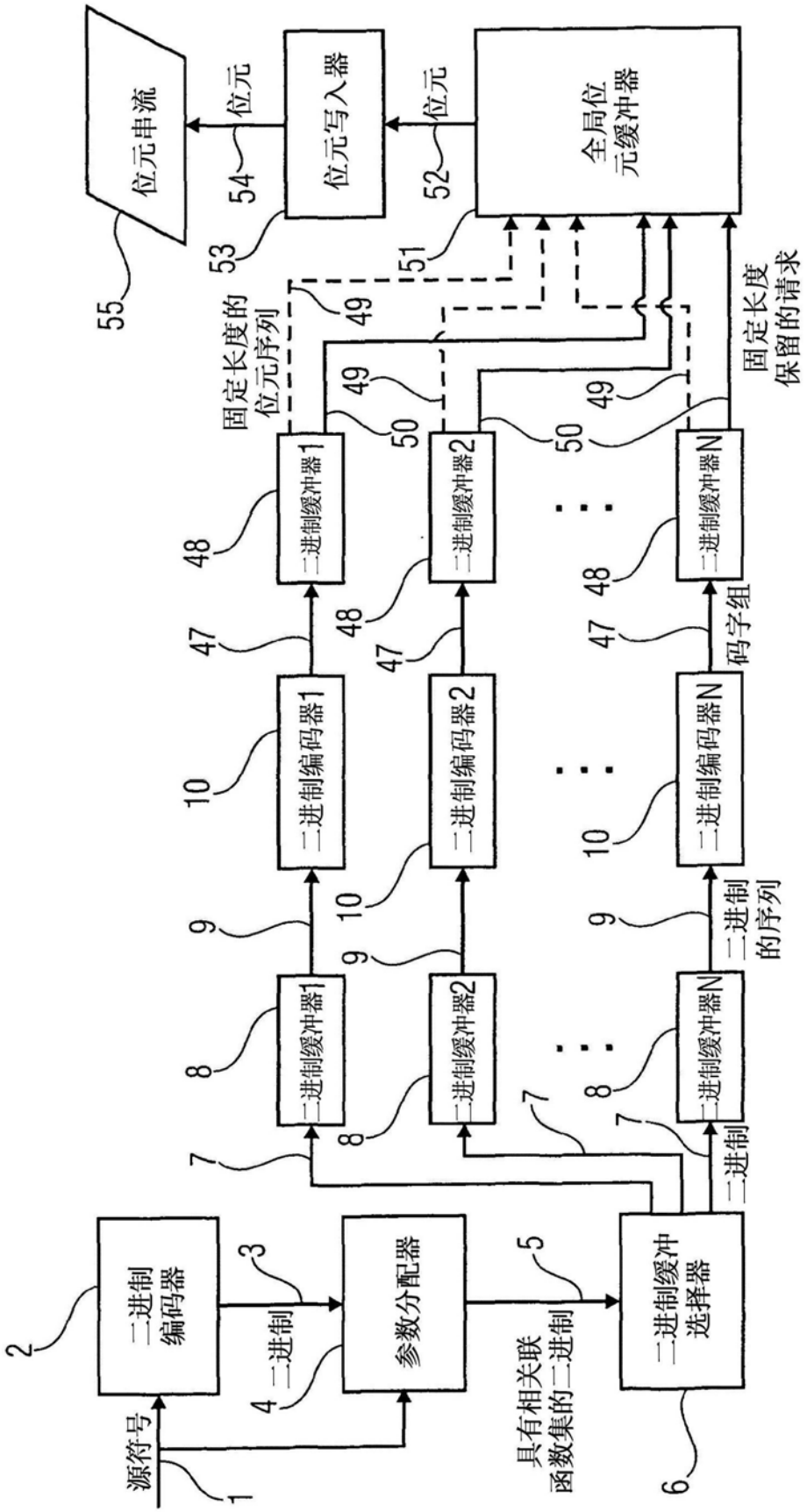

源符号
1
二进制编码器
2
二进制
4
参数分配器
3
具有相关联函数集的二进制
5
二进制缓冲选择器
6
二进制
7
二进制缓冲器1
8
二进制缓冲器2
二进制缓冲器N
二进制的序列
9
二进制编码器1
10
二进制编码器2
二进制编码器N
码字组
47
二进制缓冲器1
48
二进制缓冲器2
二进制缓冲器N
固定长度的位元序列
49
50
固定长度保留的请求
51
全局位元缓冲器
52
位元
位元写入器
54
53
位元串流
55

图11

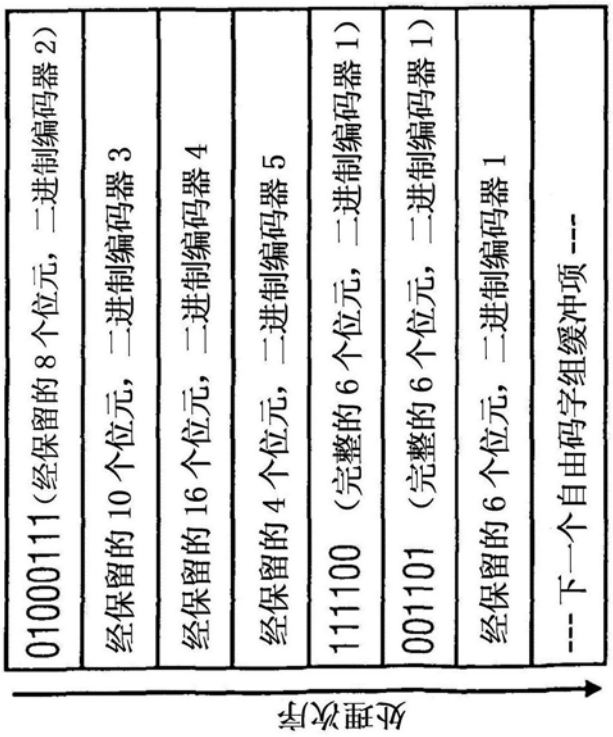

01000111(经保留的8个位元，二进制编码器2)
经保留的10个位元，二进制编码器3
经保留的16个位元，二进制编码器4
经保留的4个位元，二进制编码器5
111100 (完整的6个位元，二进制编码器1)
001101 (完整的6个位元，二进制编码器1)
经保留的6个位元，二进制编码器1
--- 下一个自由码字组缓冲项 ---
处理次序

图12A

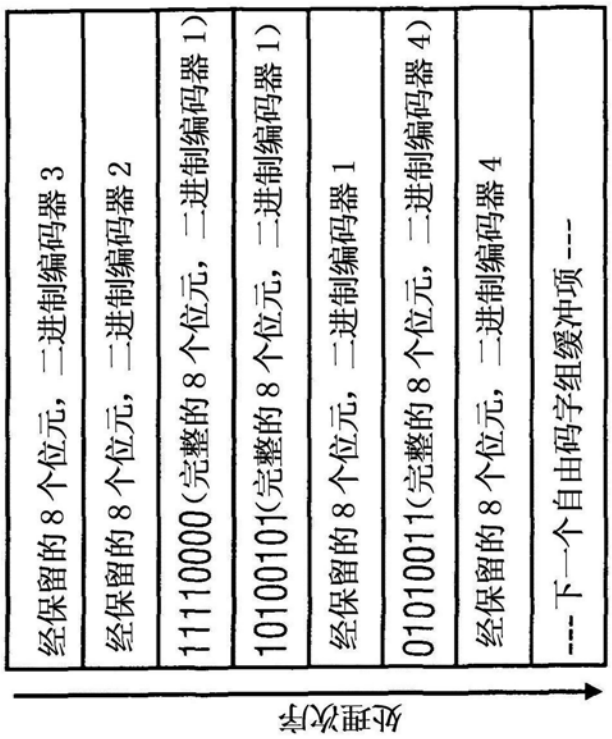

经保留的8个位元，二进制编码器3
经保留的8个位元，二进制编码器2
11110000(完整的8个位元，二进制编码器1)
10100101(完整的8个位元，二进制编码器1)
经保留的8个位元，二进制编码器1
01010011(完整的8个位元，二进制编码器4)
经保留的8个位元，二进制编码器4
--- 下一个自由码字组缓冲项 ---
处理次序

图12B

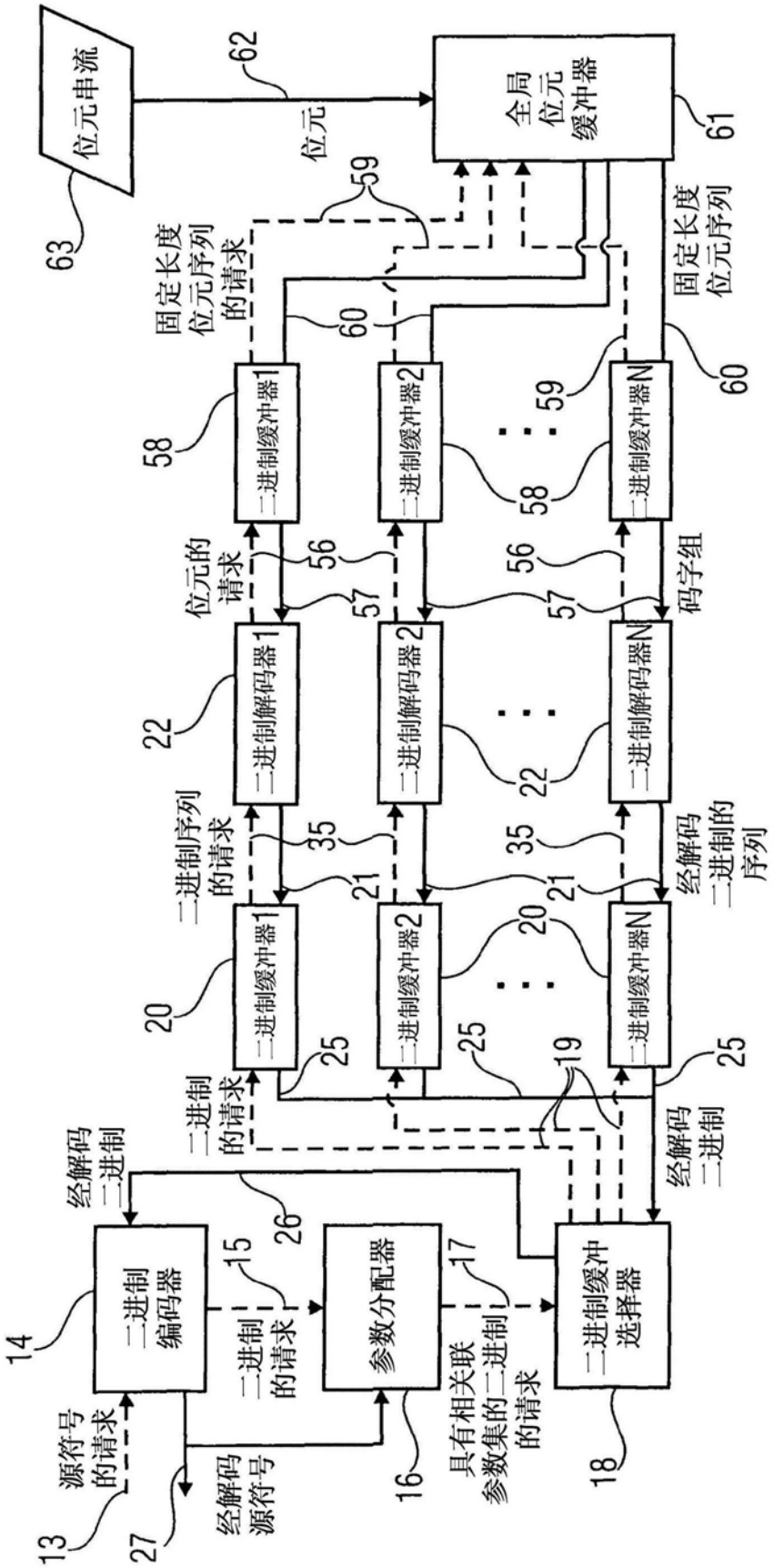

位元串流
62
位元
全局
位元
缓冲器
61
63
固定长度
位元序列
的请求
59
60
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
58
位元的
请求
56
57
码字组
二进制解码器1
二进制解码器2
二进制解码器N
22
二进制序列
的请求
35
21
经解码
二进制的
序列
20
25
19
二进制
的请求
经解码
二进制
26
二进制
编码器
14
15
二进制
的请求
参数分配器
17
具有相关联
参数集的二进制
的请求
二进制缓冲
选择器
18
源符号
的请求
13
27
经解码
源符号
16

图13

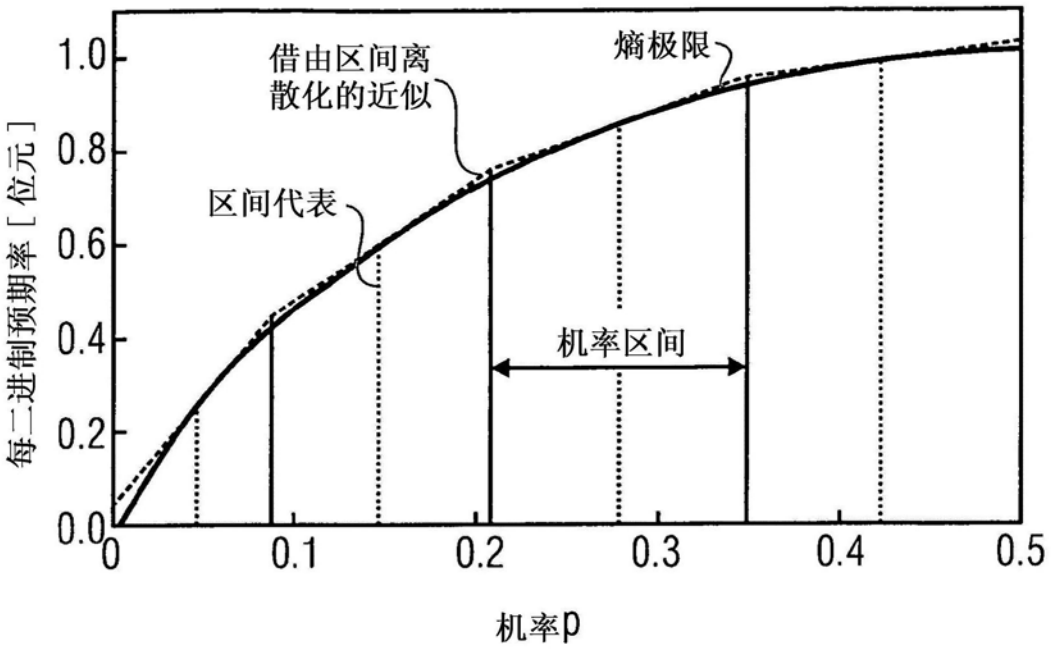

每二进制预期率[位元]
1.0
0.8
0.6
0.4
0.2
0.0
0
0.1
0.2
0.3
0.4
0.5
机率p
借由区间离
散化的近似
熵极限
区间代表
机率区间

图14

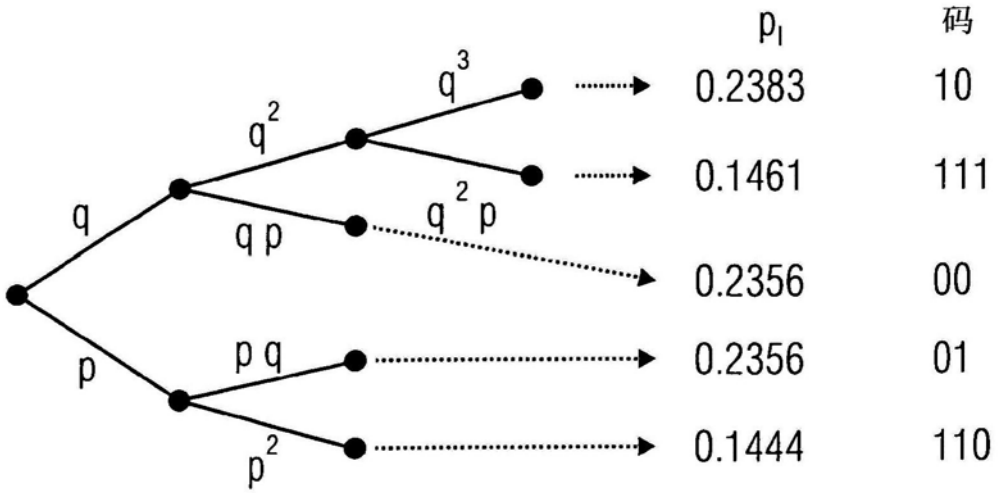

p_I
码
q^3
q^2
q
$q^2 p$
q p
p
p q
p^2
0.2383
10
0.1461
111
0.2356
00
0.2356
01
0.1444
110

图15

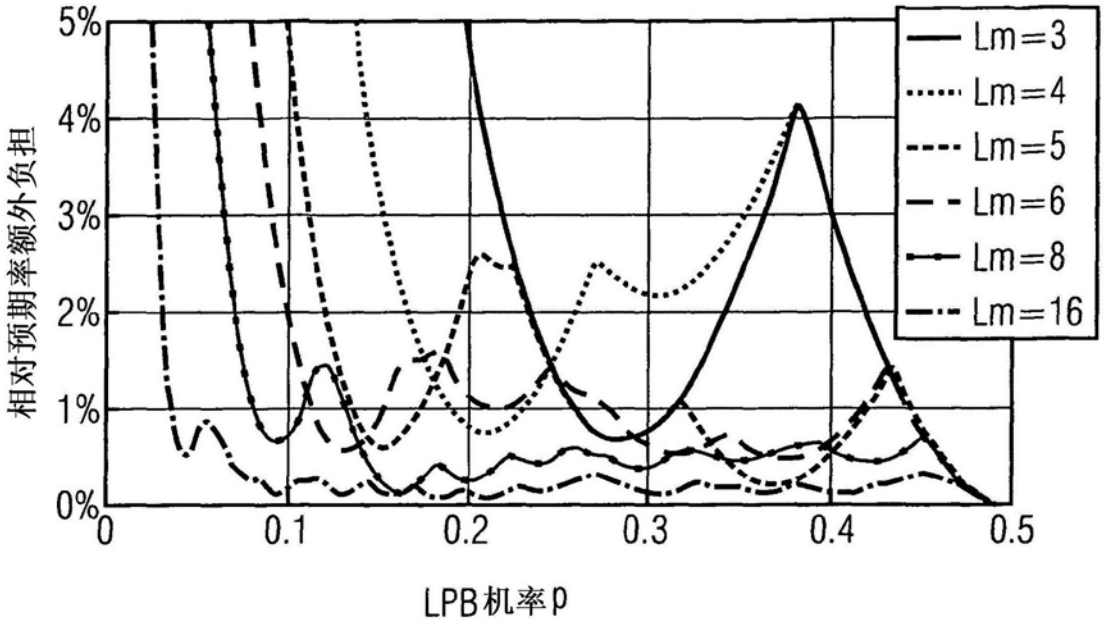

5%
4%
3%
2%
1%
0%
0
0.1
0.2
0.3
0.4
0.5
相对预期率额外负担
LPB机率p
Lm=3
Lm=4
Lm=5
Lm=6
Lm=8
Lm=16

图16

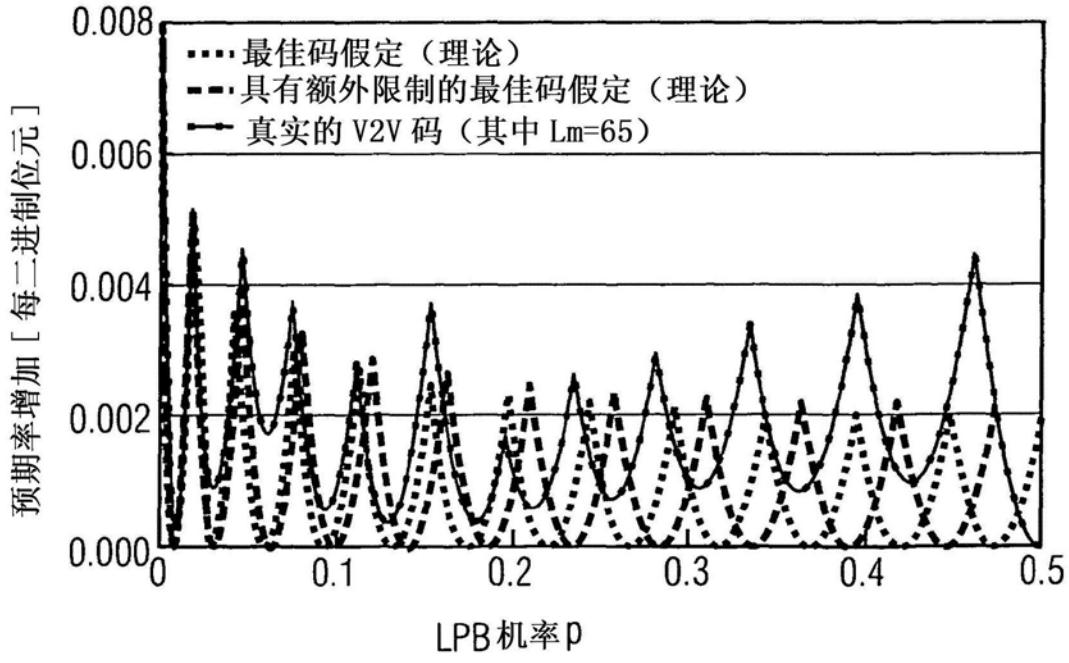

0.008
0.006
0.004
0.002
0.000
0
0.1
0.2
0.3
0.4
0.5
预期率增加［每二进制位元］
LPB机率p
最佳码假定（理论）
具有额外限制的最佳码假定（理论）
真实的 V2V 码（其中 Lm=65）

图17

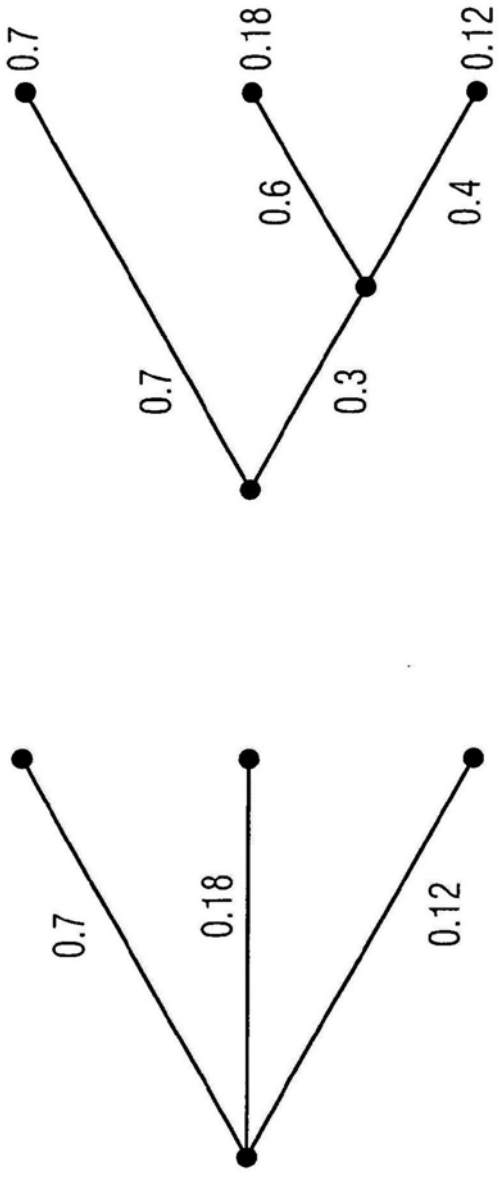


图18

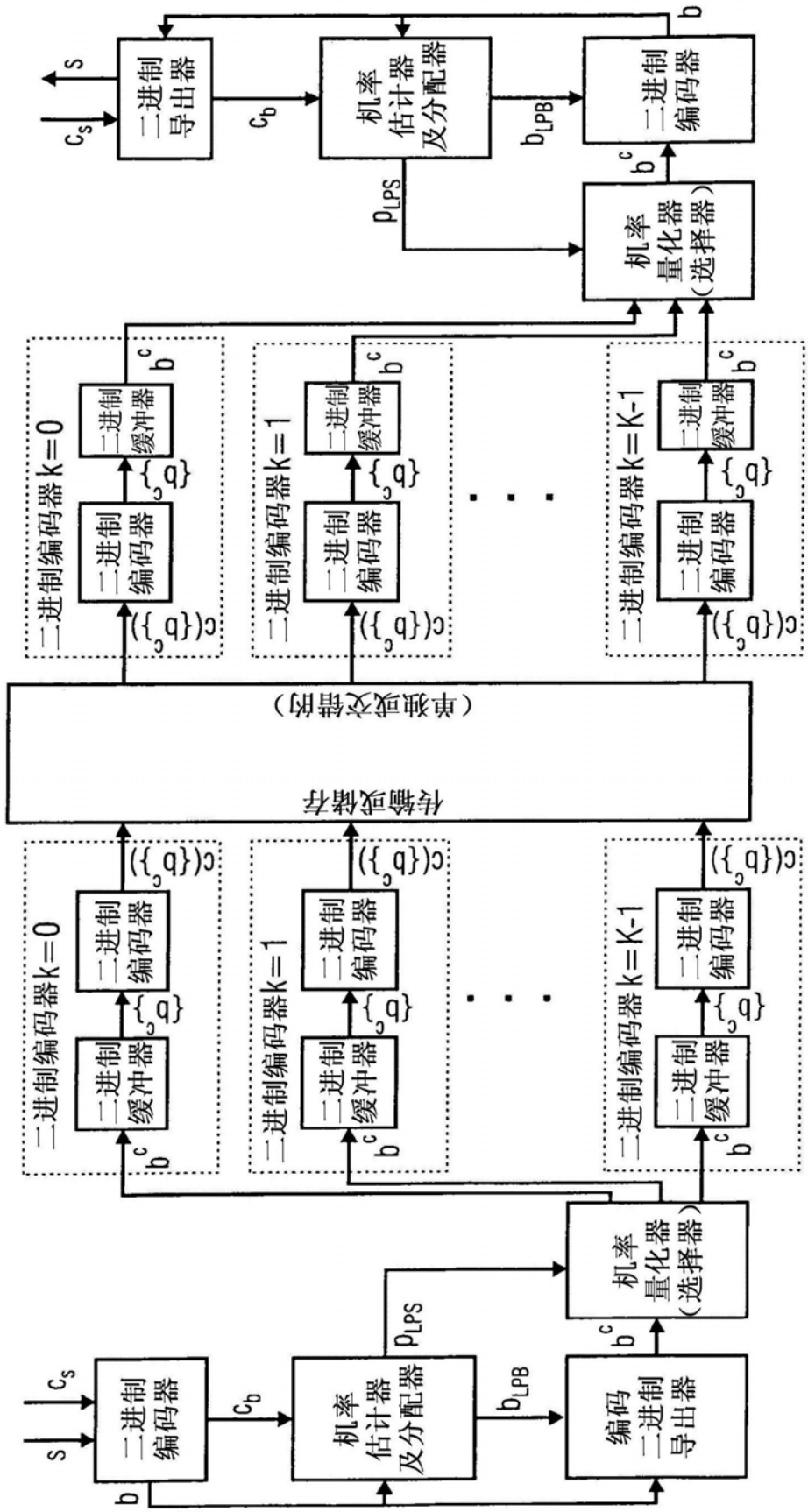

s
c_s
二进制编码器
b
c_b
机率估计器及分配器
b_{LPB}
p_{LPS}
机率量化器（选择器）
b^c
编码二进制导出器
二进制编码器k=0
二进制编码器k=1
二进制编码器k=K-1
二进制缓冲器
二进制编码器
$c(\{b^c_q\})$
$\{b^c_q\}$
传输或储存
（单独或交错的）
二进制导出器
二进制编码器

图19

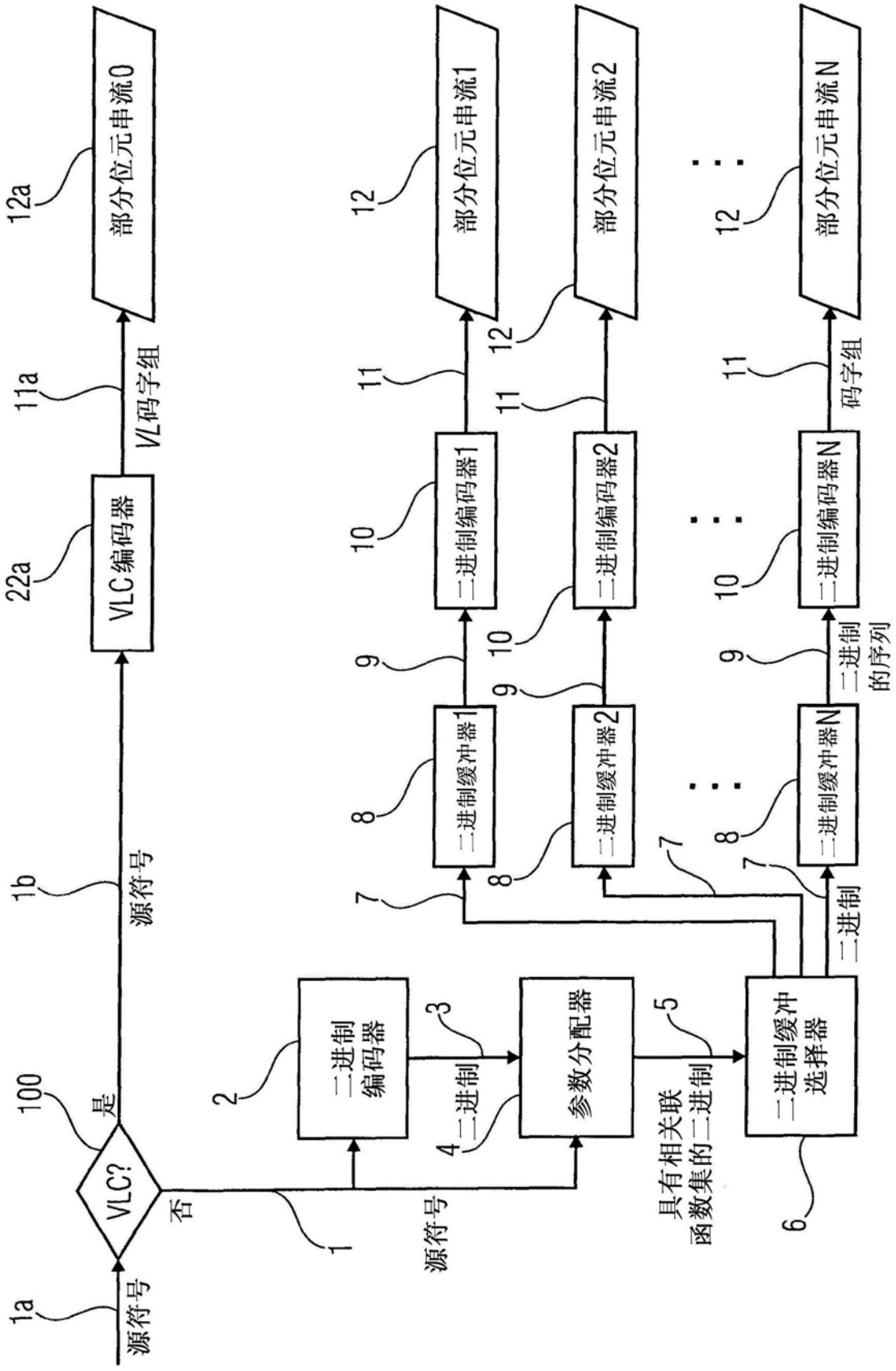

1a
源符号
100
VLC?
是
否
1b
源符号
VL码字组
22a
VLC编码器
11a
12a
部分位元串流0
1
源符号
2
二进制
编码器
3
二进制
4
参数分配器
具有相关联
函数集的二进制
5
二进制缓冲
选择器
6
二进制
7
二进制缓冲器1
8
二进制编码器1
9
10
部分位元串流1
11
12
二进制缓冲器2
二进制编码器2
部分位元串流2
二进制缓冲器N
二进制编码器N
部分位元串流N
码字组
二进制
的序列

图20

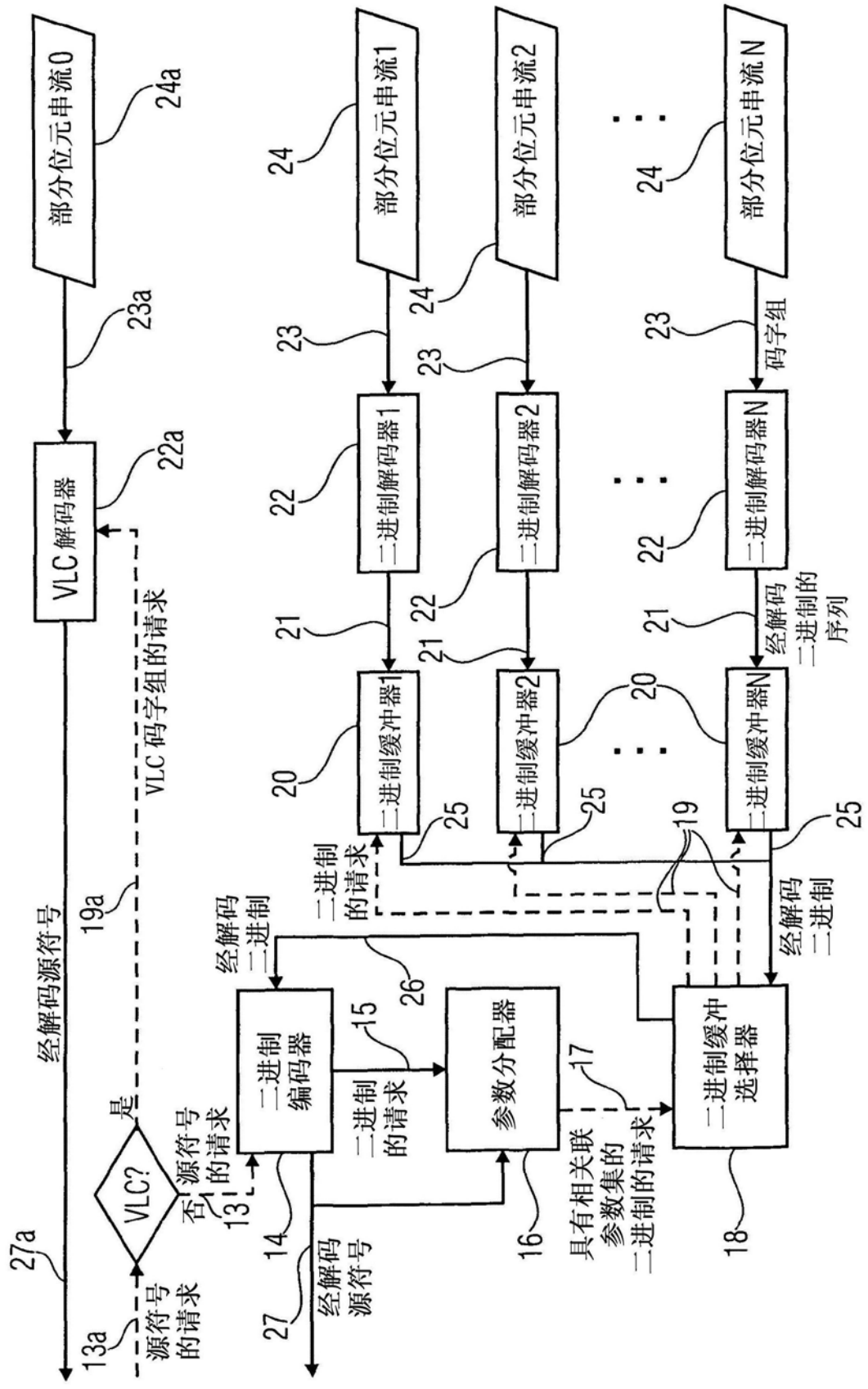

部分位元串流0
24a
23a
VLC 解码器
22a
VLC 码字组的请求
经解码源符号
19a
27a
VLC?
是
否
源符号的请求
13
13a
源符号的请求
部分位元串流1
部分位元串流2
部分位元串流N
24
23
码字组
二进制解码器1
二进制解码器2
二进制解码器N
22
21
经解码二进制的序列
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
20
25
19
二进制的请求
经解码二进制
经解码二进制
二进制编码器
15
二进制的请求
26
参数分配器
17
具有相关联参数集的二进制的请求
二进制缓冲选择器
14
16
18
27
经解码源符号

图21

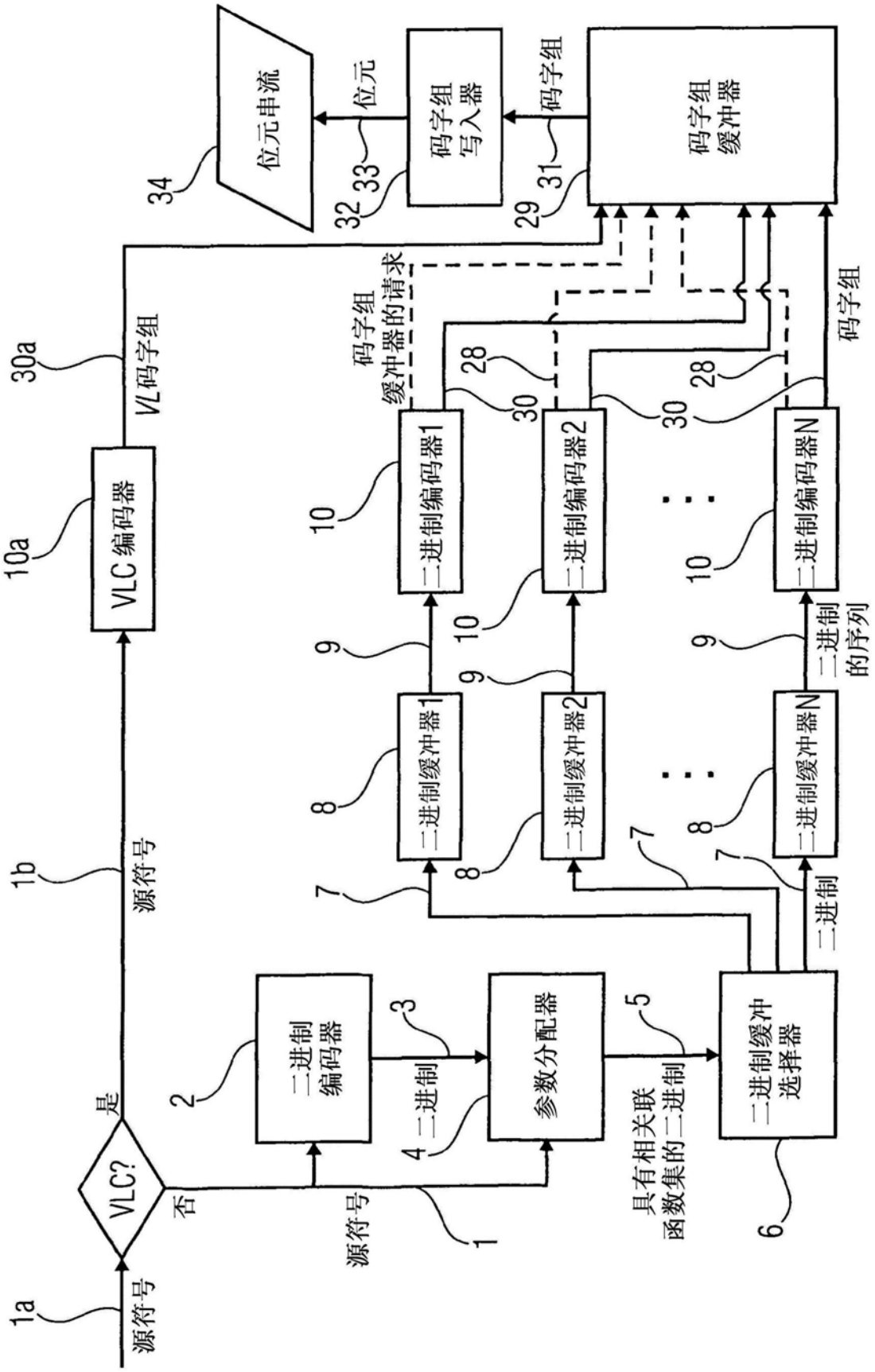

1a
源符号
VLC?
是
否
1b
源符号
VLC 编码器
10a
30a
VL码字组
二进制
编码器
2
源符号
1
3
二进制
4
参数分配器
5
具有相关联
函数集的二进制
二进制缓冲
选择器
6
二进制
二进制编码器1
二进制编码器2
二进制编码器N
10
9
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
8
7
二进制
的序列
码字组
缓冲器的请求
28
30
码字组
码字组
缓冲器
29
31
码字组
写入器
32
33
位元
位元串流
34

图22

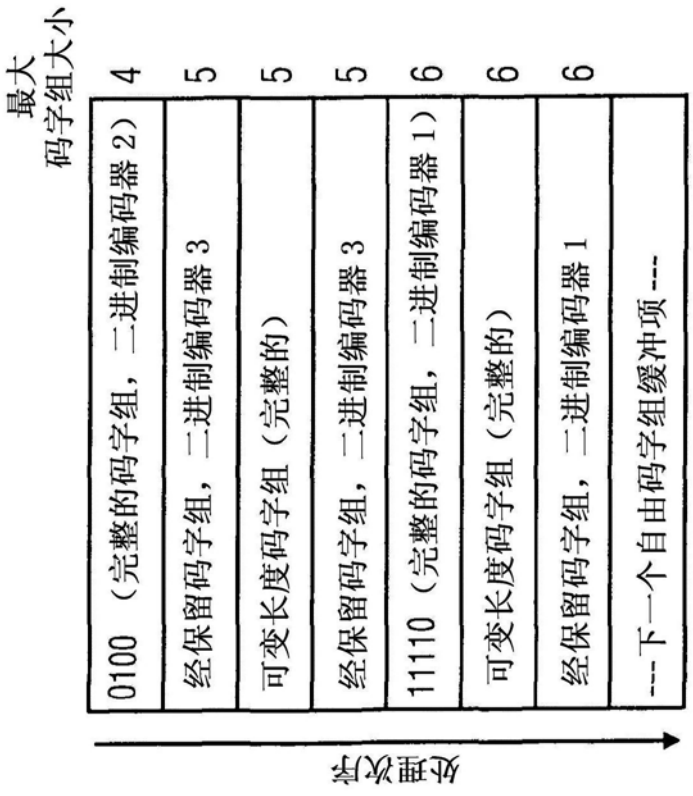

最大
码字组大小
0100 （完整的码字组，二进制编码器 2） 4
经保留码字组，二进制编码器 3 5
可变长度码字组（完整的） 5
经保留码字组，二进制编码器 3 5
11110（完整的码字组，二进制编码器 1） 6
可变长度码字组（完整的） 6
经保留码字组，二进制编码器 1 6
--- 下一个自由码字组缓冲项 ---
处理次序

图23A

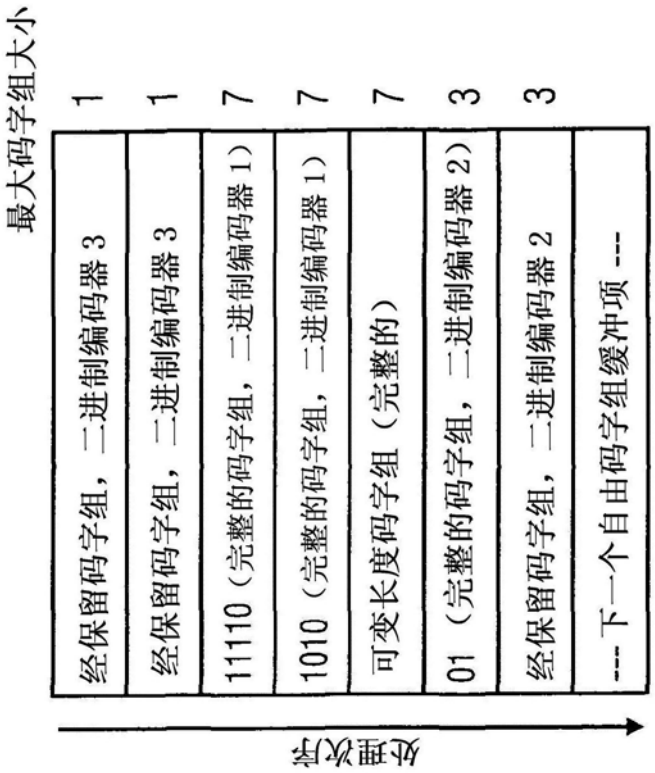

最大码字组大小
经保留码字组，二进制编码器 3 1
经保留码字组，二进制编码器 3 1
11110（完整的码字组，二进制编码器 1） 7
1010（完整的码字组，二进制编码器 1） 7
可变长度码字组（完整的） 7
01（完整的码字组，二进制编码器 2） 3
经保留码字组，二进制编码器 2 3
--- 下一个自由码字组缓冲项 ---
处理次序

图23B

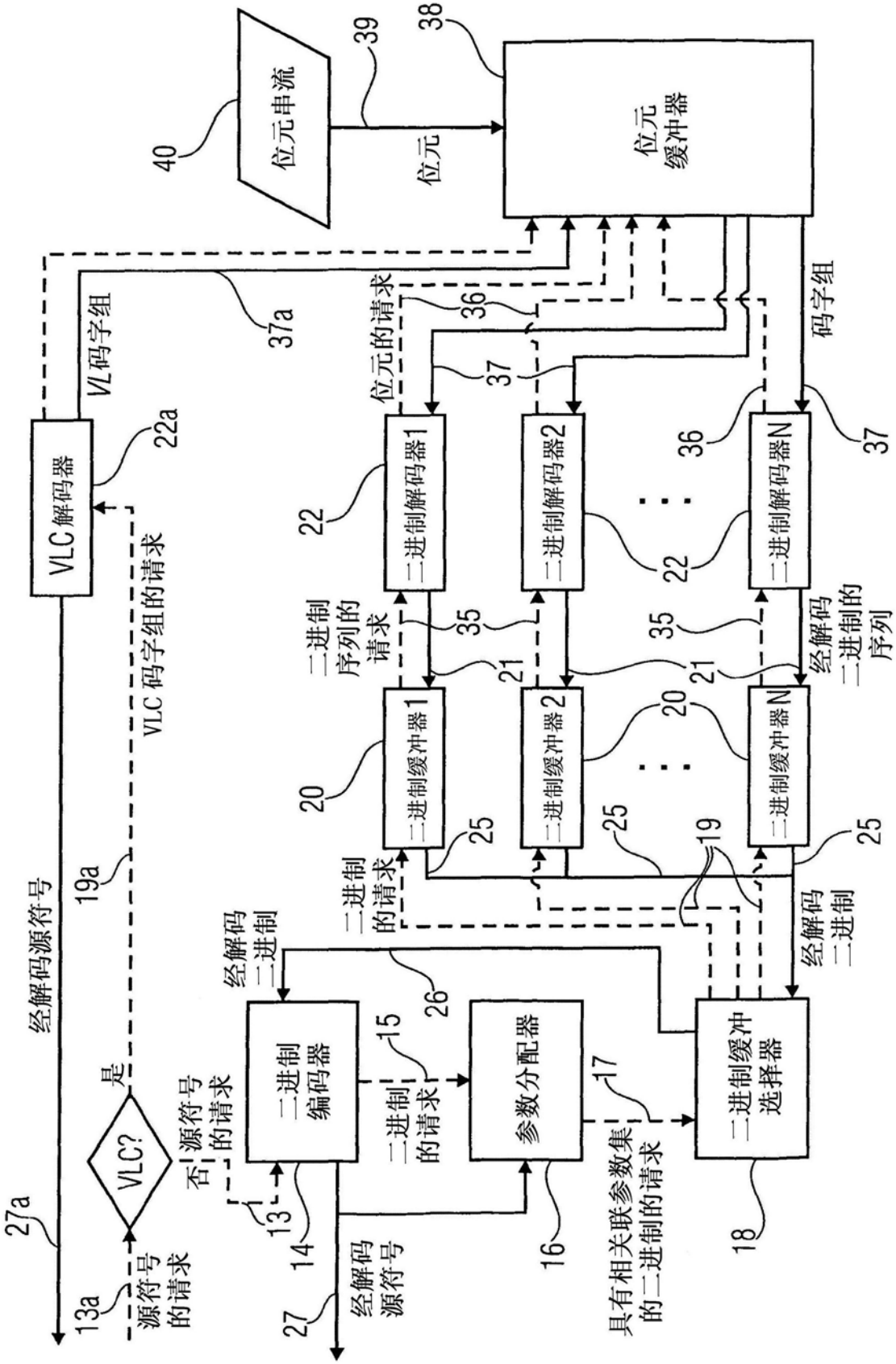

位元串流
位元
位元缓冲器
VLC解码器
VL码字组
VLC码字组的请求
位元的请求
码字组
二进制解码器1
二进制解码器2
二进制解码器N
二进制序列的请求
经解码二进制的序列
二进制缓冲器1
二进制缓冲器2
二进制缓冲器N
二进制的请求
经解码二进制
经解码源符号
二进制编码器
参数分配器
二进制缓冲选择器
二进制的请求
具有相关联参数集的二进制的请求
VLC?
是
否
源符号的请求
源符号的请求
经解码源符号
经解码源符号
13
13a
14
15
16
17
18
19
19a
20
21
22
22a
25
26
27
27a
35
36
37
37a
38
39
40

图24