

说明书

一种数据处理方法和数据处理装置

本申请要求于 2023 年 02 月 25 日提交国家知识产权局、申请号为 202310206681.6、发明名称为“一种数据处理方法和数据处理装置”的中国专利申请的优先权、于 2023 年 03 月 28 日提交国家知识产权局、申请号为 202310359894.2、发明名称为“一种数据处理方法和数据处理装置”的中国专利申请的优先权、于 2023 年 04 月 28 日提交国家知识产权局、申请号为 202310491179.4、发明名称为“一种数据处理方法和数据处理装置”的中国专利申请的优先权、于 2023 年 08 月 18 日提交国家知识产权局、申请号为 202311055979.8、发明名称为“一种数据处理方法和数据处理装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及通信领域，尤其涉及一种数据处理方法和数据处理装置。

背景技术

在 5G、云计算、大数据、人工智能等持续推动下，高速光传输网络正朝着大容量、分组化、智能化的方向发展。相干光通信系统利用光波的幅度，相位，偏振和频率来承载信息。为了对抗在传输过程中因色散、偏振相关损伤、噪声、非线性效应及其它因素引起的光信号失真并保持长距离传输，相干光通信系统通常需要采用高效的前向纠错（Forward Error Correction, FEC）码来对抗光传输过程中的光学损伤，以保持长距离传输时其误比特率足够低。如当前 400ZR+和 800ZR 采纳的开放 FEC 码（OpenFEC），简称为 OFEC 码，其冗余（overhead, OH）为 15.3%，当采用软判决译码，性能约为纠前 $2.0E-2$ 。

为了提高频谱效率，通常采用多电平的正交振幅调制（Quadrature Amplitude Modulation, QAM），如 16QAM、32QAM、64QAM 甚至更高阶 QAM。传统的 QAM 调制对应的信号星座图上的每个星座点（constellation points）都是以相同的概率出现。概率星座整形（Probabilistic Constellation Shaping, PCS）技术，在保持星座点位置不变下改变星座点出现的概率，使其非均匀分布，从而提升系统传输性能。PCS 技术作为一种调制格式优化技术，具有逼近香农极限、灵活可变等优点，已被广泛研究与应用。现有采用 OFEC 编码的数据处理和传输方法主要是采用传统 QAM 调制，无法适用未来采用 PCS 技术的场景，是未来亟需解决的问题。

发明内容

本申请实施例提供了一种数据处理方法和数据处理装置，将采用 FEC 编码并结合 PCS 技术以满足更长传输距离，并使得整体数据处理操作更简单，复杂度更低，功耗更低，提升了整体性能。

第一方面，本申请实施例提供了一种数据处理方法，该数据处理方法包括如下步骤。对 k 个比特中的第一比特集合进行 PCS 处理得到第二比特集合， k 为大于 1 的整数。对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合。所述第四比特集合包括 m_0 个第一比特子集，每个所述第一比特子集包括 F_0 个比特，所述 m_0 为大于 1 的整数， F_0 为大于 1 的偶数。对所述第四比特集合进行第一交织处理得到第五比特集合。所述第五比特集合包括 m_0 个第二比特子集，每个所述第二比特子集包括 F_0 个比特，每个所述第二比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合，每个所述第二比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在该实施方式中，对于未来城域电信传输和城域 DCI 互联场景，将采用 FEC 编码并结合 PCS 技术以满足更长传输距离。PCS 处理的引入，要求在符号映射操作时，映射到一个调制符号的符号位（sign bits）比特是 0、1 等概的，且映射到一个调制符号的幅度位（amplitude bits）是 0、1 不等概的。以 FEC 编码具体采用 OFEC 编码为例，为了尽量避免对现有 OFEC 编码、OFEC 交织器的影响，需要在 OFEC 编码之后、符号映射操作之前引入一个新的交织器，使得经过 PCS 处理得到的 0 和 1 不等概的比特可以映射到调制符号的幅度位比特，实现了在保持星座点位置不变下改变星座点出现的概率，使其非均匀分布，提

升了整体性能,以满足未来更长传输距离需求。并且,实现方式简单、复杂度低、功耗低。

在一些可能的实施方式中,每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列,所述第二比特子集的每行 F_1 个比特中的其中 $F_1/2$ 个比特来自所述第二比特集合,所述第二比特集合包括 $m_0 \times F_0/2$ 个比特,所述第二比特子集的每行 F_1 个比特中的另外 $F_1/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在一些可能的实施方式中, $F_0 = 256$, $F_1 = 16$,所述第二比特子集中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特来自所述第二比特集合, $0 \leq i_2 < 16$, $0 \leq j_2 < 8$ 。或者,所述第二比特子集中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自所述第二比特集合, $0 \leq i_2 < 16$, $0 \leq j_2 < 8$ 。其中, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

10 在一些可能的实施方式中, $F_0 = 256$,所述第四比特集合中的 m_1 个所述第一比特子集总共 $m_1 \times 256$ 个比特来自所述第二比特集合, $m_0 = m_1 \times 2$ 。

在一些可能的实施方式中, $F_0 = 256$,所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列,所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特,其中, $0 \leq r_0 < 32$, $0 \leq r_1 < 32$ 。

15 在一些可能的实施方式中,所述 m_0 个第一比特子集分布为 2 行 m_1 列,其中,第 0 行第 m_2 列的第一比特子集 B_{0,m_2} 和第 1 行第 m_2 列的第一比特子集 B_{1,m_2} 来自所述第二比特集合, $0 \leq m_2 < m_1/2$ 。

在一些可能的实施方式中,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 $\left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% (m_0 \times 8)$ 个比特来自所述第四比特集合中第 r_0 行的第 0 比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特,其中, $0 \leq \tilde{c}_0 < (m_0 \times 4)$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

20 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的

$$\text{整数倍, } c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)。$$

25 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的

$$\text{整数倍, } c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor。$$

30 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的

$$\text{整数倍, } c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times (m_0 \times 4)。$$

35 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times (m_0 \times 4)$ 。

5 在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合中第 r_0 行的第 0 比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < (m_0 \times 4)$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

10 其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = c_0 \% (m_0 \times 4) \times 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

15 其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = c_0 \% (m_0 \times 4) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

20 其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + 16 \times \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

25 其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + 16 \times \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor$ 。

30 在一些可能的实施方式中, 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合包括: 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数。对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

35 在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列的第三比特子集、第 1 列的第三比特子集、第 4 列的第三比特子集和第 5 列的第三比特子集中的比特来自所述第二比特集

合,第2列的第三比特子集、第3列的第三比特子集和第6列的第三比特子集中的比特来自所述第三比特集合。

5 在一些可能的实施方式中,所述 m_0 个第一比特子集分布为2行 m_1 列, $m_0=16$, $m_1=8$,第0行第0列的第一比特子集 $B_{0,0}$ 、第0行第1列的第一比特子集 $B_{0,1}$ 、第0行第4列的第一比特子集 $B_{0,4}$ 、第0行第5列的第一比特子集 $B_{0,5}$ 、第1行第0列的第一比特子集 $B_{1,0}$ 、第1行第1列的第一比特子集 $B_{1,1}$ 、第1行第4列的第一比特子集 $B_{1,4}$ 和第1行第5列的第一比特子集 $B_{1,5}$ 来自所述第二比特集合。

在一些可能的实施方式中,所述第五比特集合中第 r_1 行的128个比特中第 $\left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 128$ 个比特来自所述第四比特集合中第 r_0 行的第0比特到第31比特或第64比特到第95比特中的一个比特,其中, $0 \leq \tilde{c}_0 < 64$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

10 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

15 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。在一些可能的实施方式中,所述第五比特集

20 合中第 r_1 行的128个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

25 在一些可能的实施方式中,所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍,且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

30 在一些可能的实施方式中,第五比特集合中第 r_1 行的128个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合中第 r_0 行的第0比特到第31比特或第64比特到第95比特中的一个比特,其中, $0 \leq \tilde{c}_0 < 64$ 。

在一些可能的实施方式中,第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 32) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right)\right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$5 \quad c_1 = (c \% \theta + (r_1 + 1 + 2 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 32) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right)\right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$10 \quad c_1 = (c \% \theta + (r_1 + 2 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right)\right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$15 \quad c_1 = (c \% \theta + (r_1 + 1 + 2 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 3 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 3 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 4 列至第 5 列的每个第三比特子集中的 F_0 个比特来自所述第二比特集合, 第 6 列的每个第三比特子集中的 F_2 个比特来自所述第三比特集合。

在一些可能的实施方式中, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列的比特来自所述第二比特集合。或者, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $(j_1 \times 2)$ 列的比特来自所述第二比特集合。其中, $0 \leq i_0 < 2$, $0 \leq j_0 < 4$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$, 第 0 列至第 3 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 3 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 4 列至第 5 列的每个第一比特子集中的 F_0 个比特来自所述第二比特集合, 第 6 列至第 7 列的每个第一比特子集中的 F_0 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在一些可能的实施方式中, 对所述第四比特集合进行第一交织处理得到第五比特集合包括: 对所述第四比特集合中第 4 列至第 7 列的第一比特子集进行第一交织处理得到对应的所述第五比特集合中第 4 列至第 7 列的第二比特子集。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 128 个比特中第 $64 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + \right)$

$(r_1 + 2) \% 4$) % 64 个比特来自所述第四比特集合中第 r_0 行的第 64 比特到第 95 比特中的一个比特, 其中,

$0 \leq \tilde{c}_0 < 31$, $\lfloor a \rfloor$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

5 在一些可能的实施方式中, 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$;

或者,

10 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

15 其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$;

或者,

当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

20
$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$;

或者,

25 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

30 在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 0 行第 5 列的第三比特子集 $I_{0,5}$ 、第 1 行第 5 列的第三比特子集 $I_{1,5}$ 、第 0 行第 6 列的第三比特子集 $I_{0,6}$ 中第 0 列至第 7 列的比特和第 1 行第 6 列的第三比特子集 $I_{1,6}$ 中第 0 列至第 7 列的比特来自所述第二比特集合, 第 0 行第 6 列的第三比特子集 $I_{0,6}$ 中第 8 列至第 14 列的比特和第 1 行第 6 列的第三比特子集 $I_{1,6}$ 中第 8 列至第 14 列的比特来自所述第三比特集合。

35

在一些可能的实施方式中, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列的比特来自所述第二比特集合。或者, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $(j_1 \times 2)$ 列的比特来自所述第二比特集合。其中, $0 \leq i_0 < 2$, $0 \leq j_0 < 5$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

- 5 在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$, 第 0 列至第 4 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合。第 0 行第 5 列的第一比特子集 $B_{0,5}$ 、第 1 行第 5 列的第一比特子集 $B_{1,5}$ 、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 0 行至第 7 行第 0 列至第 7 列的 8×8 个比特、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 8 行至第 15 行第 8 列至第 15 列的 8×8 个比特、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 0 行至第 7 行第 0 列至第 7 列的 8×8 个比特和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 8 行至第 15 行第 8 列至第 15 列的 8×8 个比特来自所述第二比特集合。第 0 行第 7 列的第一比特子集 $B_{0,7}$ 、第 1 行第 7 列的第一比特子集 $B_{1,7}$ 、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 0 行至第 7 行第 8 列至第 15 列的 8×8 个比特、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 8 行至第 15 行第 0 列至第 7 列的 8×8 个比特、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 0 行至第 7 行第 8 列至第 15 列的 8×8 个比特和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 8 行至第 15 行第 0 列至第 7 列的 8×8 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在一些可能的实施方式中, 对所述第四比特集合进行第一交织处理得到第五比特集合包括: 对所述第四比特集合中第 5 列至第 7 列的第一比特子集进行第一交织处理得到对应的所述第五比特集合中第 5 列至第 7 列的第二比特子集。

- 20 在一些可能的实施方式中, 当 $0 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 8$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 $80 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 48$ 个比特来自所述第四比特集合中第 r_0 行的第 80 比特到第 103 比特中的一个比特。当 $8 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 16$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 $80 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 48$ 个比特来自所述第四比特集合中第 r_0 行的第 80 比特到第 95 比特和第 104 比特到第 111 比特中的一个比特。其中, $0 \leq \tilde{c}_0 < 24$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。
- 25 在一些可能的实施方式中, 当 $0 \leq c_1 < 80$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $80 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特;

$$c_1 = 80 + (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

- 30 其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍, 且 48 能被 θ 整除, Δ 为整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(1 - \left\lfloor \frac{c_2}{24} \right\rfloor\right) \times 2, c_2 = (r_0 \% 16) \wedge c_0;$$

或者,

- 当 $0 \leq c_1 < 80$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $80 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特;

$$c_1 = 80 + (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍, 且 48 能被 θ 整除, Δ 为整数, β 是 4 的整数,

$$c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(\left\lfloor \frac{c_2}{24} \right\rfloor \right) \times 2, c_2 = (r_0 \% 16) \wedge c_0.$$

在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$ 。第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特。第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 5 个比特来自所述第三比特集合。第 5 列的每个第三比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第三比特子集中每行的另外 7 个比特来自所述第三比特集合。第 6 列的每个第三比特子集中的 F_2 个比特来自所述第二比特集合。

在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$ 。第 0 列至第 4 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合。第 5 列的每个第一比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第一比特子集中每行的另外 7 个比特来自所述第三比特集合。第 6 列的每个第一比特子集中第 i_1 行第 $15 - i_1$ 列的比特来自所述 FEC 编码的校验比特, 第 6 列的每个第一比特子集中除了第 i_1 行第 $15 - i_1$ 列的比特之外的其他比特来自所述第二比特集合, $0 \leq i_1 \leq 15$ 。第 7 列的每个第一比特子集中的 F_0 个比特来自所述 FEC 编码的校验比特。

在一些可能的实施方式中, 对所述第四比特集合进行第一交织处理得到第五比特集合包括: 对第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中第 C 列至第 15 列、第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中第 C 列至第 15 列、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 、第 0 行第 7 列的第一比特子集 $B_{0,7}$ 、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 进行第一交织处理得到对应的第 0 行第 5 列的第二比特子集 $T_{0,5}$ 中第 C 列至第 15 列、第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中第 C 列至第 15 列、第 0 行第 6 列的第二比特子集 $T_{0,6}$ 、第 0 行第 7 列的第二比特子集 $T_{0,7}$ 、第 1 行第 6 列的第二比特子集 $T_{1,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 。其中, $0 \leq C \leq 15$ 。

在一些可能的实施方式中, $r_0 = r_1$ 。

在一些可能的实施方式中, 每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列, 所述第二比特子集中每行 F_1 个比特来自 2 个所述第一比特子集。

在一些可能的实施方式中, 每个所述第一比特子集中的 F_0 个比特分布为 F_1 行 F_1 列, 所述第二比特子集中每行 F_1 个比特的其中 $F_1/2$ 个比特来自其中一个第一比特子集的其中一行, 所述第二比特子集中每行 F_1 个比特的另外 $F_1/2$ 个比特来自另外一个第一比特子集的其中一行。

在一些可能的实施方式中, $F_0 = 256$, 所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中每列 32 个比特来自 2 个所述第一比特子集。

在一些可能的实施方式中, 每个所述第一比特子集中的 256 个比特分布为 16 行 16 列, 所述第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的其中一行, 所述第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的其中一行。

在一些可能的实施方式中, 所述第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的第 $c_1 \% 16$ 列, 所述第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的第 $c_1 \% 16$ 列, 其中, $0 \leq c_1 < (m_0 \times 8)$ 。

在一些可能的实施方式中, $m_0=16$, 所述 16 个第二比特子集分布为 2 行 8 列;

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特,

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特,

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特,

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特,

5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

在一些可能的实施方式中, $m_0=12$ 或 16。

在一些可能的实施方式中, 对所述第四比特集合进行交织处理得到第五比特集合之后, 所述方法还包括: 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 L/2 条第二比特流, 每条所述第
5 一比特流包括多个所述第五比特集合, L 为大于 0 的偶数。对 L/2 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数。对所述双偏振符号流进行数字信号处理 (Digital Signal Processing, DSP) 成帧。

在一些可能的实施方式中, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

10 在一些可能的实施方式中, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

在一些可能的实施方式中, 对所述双偏振符号流进行 DSP 成帧包括: 将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

第二方面, 本申请提供了一种数据处理方法, 该数据处理方法包括如下步骤。从数据帧中获取第一
15 数据, 所述第一数据包括 r 行 q 列个比特, 所述 r 为大于 0 的整数, 所述 q 为大于 0 的整数。对所述第一数据进行循环冗余校验 CRC 和/或插入填充比特得到第二数据, 所述第二数据包括进行 CRC 插入的 d_{CRC} 个校验比特和/或 d_{PAD} 个填充比特, d_{CRC} 为大于或等于 0 的整数, d_{PAD} 为大于或等于 0 的整数。对所述第二数据进行加扰得到第三数据, 所述第三数据的比特数量是 $d_{scr} = r \times q + d_{CP}$, $d_{CP} = d_{CRC} + d_{PAD}$ 。获取所述第三数据中的 k 个比特, k 为大于 1 的整数, d_{scr} 是 k 的整数倍。对 k 个比特中的第一比特集
20 合进行 PCS 处理得到第二比特集合。对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合。

在该实施方式中, 对于未来城域电信传输和城域 DCI 互联场景, 将采用 FEC 编码并结合 PCS 技术以满足更长传输距离。PCS 处理的引入, 使得进行 PCS 处理和 FEC 编码的信息比特长度不再是 3552, 而是更小的数值, 此时需要重新设计从数据帧获取数据的行数, 进行 CRC 校验和插入填充比特的数量, 以使
25 得整体数据处理操作更简单, 复杂度更低, 功耗更低。

在一些可能的实施方式中, 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合包括: 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2
30 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数。对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

在一些可能的实施方式中, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

35 在一些可能的实施方式中, d_{scr} 是 $4 \times k$ 的整数倍, $q=10280$ 。

在一些可能的实施方式中, 所述进行 CRC 和/或插入填充比特的冗余为 $(d_{CRC} + d_{PAD})/(r \times q)$, $(d_{CRC} + d_{PAD})/(r \times q) \leq 0.001$ 。

在一些可能的实施方式中, $r=79$, $d_{CP}=328$, $k=3224$;

或者, $r=83$, $d_{CP}=32$, $k=3386$;

40 或者, $r=83$, $d_{CP}=536$, $k=3388$;

或者, $r=87$, $d_{CP}=72$, $k=2662$;

或者, $r=87$, $d_{CP}=744$, $k=2664$;

或者, $r=87$, $d_{CP}=3432$, $k=2672$;

或者, $r=104$, $d_{CP}=704$, $k=3184$;

45 或者, $r=104$, $d_{CP}=3392$, $k=3192$;

或者, $r=105$, $d_{CP}=504$, $k=3214$;

- 或者, $r=105$, $d_{CP}=1176$, $k=3216$;
或者, $r=109$, $d_{CP}=40$, $k=2668$;
或者, $r=109$, $d_{CP}=880$, $k=2670$;
或者, $r=110$, $d_{CP}=176$, $k=3366$;
5 或者, $r=110$, $d_{CP}=848$, $k=3368$;
或者, $r=110$, $d_{CP}=3536$, $k=3376$;
或者, $r=111$, $d_{CP}=648$, $k=3398$;
或者, $r=131$, $d_{CP}=8$, $k=2672$;
或者, $r=131$, $d_{CP}=1016$, $k=2674$;
10 或者, $r=131$, $d_{CP}=680$, $k=3208$;
或者, $r=132$, $d_{CP}=480$, $k=3232$;
或者, $r=132$, $d_{CP}=1320$, $k=3234$;
或者, $r=137$, $d_{CP}=320$, $k=3354$;
或者, $r=137$, $d_{CP}=1160$, $k=3356$;
15 或者, $r=138$, $d_{CP}=120$, $k=3378$;
或者, $r=138$, $d_{CP}=960$, $k=3380$;
或者, $r=152$, $d_{CP}=344$, $k=2658$;
或者, $r=152$, $d_{CP}=1520$, $k=2660$;
或者, $r=153$, $d_{CP}=648$, $k=2676$;
20 或者, $r=157$, $d_{CP}=856$, $k=3204$;
或者, $r=158$, $d_{CP}=656$, $k=3224$;
或者, $r=165$, $d_{CP}=264$, $k=3366$;
或者, $r=165$, $d_{CP}=1272$, $k=3368$;
或者, $r=166$, $d_{CP}=64$, $k=3386$;
25 或者, $r=166$, $d_{CP}=1072$, $k=3388$;
或者, $r=174$, $d_{CP}=144$, $k=2662$;
或者, $r=174$, $d_{CP}=1488$, $k=2664$;
或者, $r=175$, $d_{CP}=616$, $k=2678$;
或者, $r=183$, $d_{CP}=360$, $k=3200$;
30 或者, $r=183$, $d_{CP}=1536$, $k=3202$;
或者, $r=184$, $d_{CP}=664$, $k=3218$;
或者, $r=184$, $d_{CP}=1840$, $k=3220$;
或者, $r=185$, $d_{CP}=968$, $k=3236$;
或者, $r=192$, $d_{CP}=744$, $k=3358$;
35 或者, $r=192$, $d_{CP}=1920$, $k=3360$;
或者, $r=193$, $d_{CP}=1048$, $k=3376$;
或者, $r=194$, $d_{CP}=176$, $k=3392$;
或者, $r=194$, $d_{CP}=1352$, $k=3394$;
或者, $r=196$, $d_{CP}=616$, $k=2666$;
40 或者, $r=209$, $d_{CP}=536$, $k=3198$;
或者, $r=209$, $d_{CP}=1880$, $k=3200$;
或者, $r=210$, $d_{CP}=1008$, $k=3214$;
或者, $r=211$, $d_{CP}=136$, $k=3228$;
或者, $r=211$, $d_{CP}=1480$, $k=3230$;
45 或者, $r=217$, $d_{CP}=280$, $k=2656$;
或者, $r=217$, $d_{CP}=1960$, $k=2658$;

- 或者, $r=218$, $d_{CP}=80$, $k=2668$;
或者, $r=218$, $d_{CP}=1760$, $k=2670$;
或者, $r=219$, $d_{CP}=1224$, $k=3352$;
或者, $r=220$, $d_{CP}=352$, $k=3366$;
5 或者, $r=220$, $d_{CP}=1696$, $k=3368$;
或者, $r=221$, $d_{CP}=824$, $k=3382$;
或者, $r=221$, $d_{CP}=2168$, $k=3384$;
或者, $r=222$, $d_{CP}=1296$, $k=3398$;
或者, $r=235$, $d_{CP}=376$, $k=3196$;
10 或者, $r=235$, $d_{CP}=1888$, $k=3198$;
或者, $r=236$, $d_{CP}=680$, $k=3210$;
或者, $r=236$, $d_{CP}=2192$, $k=3212$;
或者, $r=237$, $d_{CP}=984$, $k=3224$;
或者, $r=238$, $d_{CP}=1288$, $k=3238$;
15 或者, $r=239$, $d_{CP}=920$, $k=2660$;
或者, $r=240$, $d_{CP}=1728$, $k=2672$;
或者, $r=247$, $d_{CP}=1000$, $k=3360$;
或者, $r=247$, $d_{CP}=2512$, $k=3362$;
或者, $r=248$, $d_{CP}=1304$, $k=3374$;
20 或者, $r=249$, $d_{CP}=96$, $k=3386$;
或者, $r=249$, $d_{CP}=1608$, $k=3388$;
或者, $r=250$, $d_{CP}=400$, $k=3400$;
或者, $r=250$, $d_{CP}=1912$, $k=3402$;
或者, $r=261$, $d_{CP}=216$, $k=2662$;
25 或者, $r=261$, $d_{CP}=2232$, $k=2664$;
或者, $r=261$, $d_{CP}=1560$, $k=3196$;
或者, $r=262$, $d_{CP}=16$, $k=2672$;
或者, $r=262$, $d_{CP}=2032$, $k=2674$;
或者, $r=262$, $d_{CP}=1360$, $k=3208$;
30 或者, $r=263$, $d_{CP}=1160$, $k=3220$;
或者, $r=264$, $d_{CP}=960$, $k=3232$;
或者, $r=264$, $d_{CP}=2640$, $k=3234$;
或者, $r=274$, $d_{CP}=640$, $k=3354$;
或者, $r=274$, $d_{CP}=2320$, $k=3356$;
35 或者, $r=275$, $d_{CP}=440$, $k=3366$;
或者, $r=275$, $d_{CP}=2120$, $k=3368$;
或者, $r=276$, $d_{CP}=240$, $k=3378$;
或者, $r=276$, $d_{CP}=1920$, $k=3380$;
或者, $r=277$, $d_{CP}=40$, $k=3390$;
40 或者, $r=277$, $d_{CP}=1720$, $k=3392$;
或者, $r=283$, $d_{CP}=2032$, $k=2666$;
或者, $r=284$, $d_{CP}=488$, $k=2674$;
或者, $r=284$, $d_{CP}=2672$, $k=2676$;
或者, $r=287$, $d_{CP}=896$, $k=3194$;
45 或者, $r=287$, $d_{CP}=2744$, $k=3196$;
或者, $r=288$, $d_{CP}=1704$, $k=3206$;

- 或者, $r=289$, $d_{CP}=664$, $k=3216$;
或者, $r=289$, $d_{CP}=2512$, $k=3218$;
或者, $r=290$, $d_{CP}=1472$, $k=3228$;
或者, $r=291$, $d_{CP}=432$, $k=3238$;
5 或者, $r=291$, $d_{CP}=2280$, $k=3240$;
或者, $r=302$, $d_{CP}=80$, $k=3360$;
或者, $r=302$, $d_{CP}=1928$, $k=3362$;
或者, $r=303$, $d_{CP}=888$, $k=3372$;
或者, $r=303$, $d_{CP}=2736$, $k=3374$;
10 或者, $r=304$, $d_{CP}=688$, $k=2658$;
或者, $r=304$, $d_{CP}=3040$, $k=2660$;
或者, $r=304$, $d_{CP}=1696$, $k=3384$;
或者, $r=305$, $d_{CP}=2168$, $k=2668$;
或者, $r=305$, $d_{CP}=656$, $k=3394$;
15 或者, $r=305$, $d_{CP}=2504$, $k=3396$;
或者, $r=306$, $d_{CP}=1296$, $k=2676$;
或者, $r=313$, $d_{CP}=1912$, $k=3194$;
或者, $r=314$, $d_{CP}=1712$, $k=3204$;
或者, $r=315$, $d_{CP}=1512$, $k=3214$;
20 或者, $r=316$, $d_{CP}=1312$, $k=3224$;
或者, $r=317$, $d_{CP}=1112$, $k=3234$;
或者, $r=317$, $d_{CP}=3128$, $k=3236$;
或者, $r=326$, $d_{CP}=320$, $k=2660$;
或者, $r=326$, $d_{CP}=2840$, $k=2662$;
25 或者, $r=327$, $d_{CP}=120$, $k=2668$;
或者, $r=327$, $d_{CP}=2640$, $k=2670$;
或者, $r=328$, $d_{CP}=2440$, $k=2678$;
或者, $r=329$, $d_{CP}=728$, $k=3356$;
或者, $r=329$, $d_{CP}=2744$, $k=3358$;
30 或者, $r=330$, $d_{CP}=528$, $k=3366$;
或者, $r=330$, $d_{CP}=2544$, $k=3368$;
或者, $r=331$, $d_{CP}=328$, $k=3376$;
或者, $r=331$, $d_{CP}=2344$, $k=3378$;
或者, $r=332$, $d_{CP}=128$, $k=3386$;
35 或者, $r=332$, $d_{CP}=2144$, $k=3388$;
或者, $r=333$, $d_{CP}=1944$, $k=3398$;
或者, $r=339$, $d_{CP}=744$, $k=3192$;
或者, $r=339$, $d_{CP}=2928$, $k=3194$;
或者, $r=340$, $d_{CP}=1384$, $k=3202$;
40 或者, $r=341$, $d_{CP}=2024$, $k=3212$;
或者, $r=342$, $d_{CP}=480$, $k=3220$;
或者, $r=342$, $d_{CP}=2664$, $k=3222$;
或者, $r=343$, $d_{CP}=1120$, $k=3230$;
或者, $r=343$, $d_{CP}=3304$, $k=3232$;
45 或者, $r=344$, $d_{CP}=1760$, $k=3240$;
或者, $r=348$, $d_{CP}=288$, $k=2662$;

- 或者, $r=348$, $d_{CP}=2976$, $k=2664$;
或者, $r=349$, $d_{CP}=760$, $k=2670$;
或者, $r=349$, $d_{CP}=3448$, $k=2672$;
或者, $r=350$, $d_{CP}=1232$, $k=2678$;
5 或者, $r=356$, $d_{CP}=704$, $k=3352$;
或者, $r=356$, $d_{CP}=2888$, $k=3354$;
或者, $r=357$, $d_{CP}=1344$, $k=3362$;
或者, $r=357$, $d_{CP}=3528$, $k=3364$;
或者, $r=358$, $d_{CP}=1984$, $k=3372$;
10 或者, $r=359$, $d_{CP}=440$, $k=3380$;
或者, $r=359$, $d_{CP}=2624$, $k=3382$;
或者, $r=360$, $d_{CP}=1080$, $k=3390$;
或者, $r=360$, $d_{CP}=3264$, $k=3392$;
或者, $r=361$, $d_{CP}=1720$, $k=3400$;
15 或者, $r=365$, $d_{CP}=1592$, $k=3192$;
或者, $r=366$, $d_{CP}=720$, $k=3200$;
或者, $r=366$, $d_{CP}=3072$, $k=3202$;
或者, $r=367$, $d_{CP}=2200$, $k=3210$;
或者, $r=368$, $d_{CP}=1328$, $k=3218$;
20 或者, $r=368$, $d_{CP}=3680$, $k=3220$;
或者, $r=369$, $d_{CP}=2304$, $k=2658$;
或者, $r=369$, $d_{CP}=456$, $k=3226$;
或者, $r=369$, $d_{CP}=2808$, $k=3228$;
或者, $r=370$, $d_{CP}=592$, $k=2664$;
25 或者, $r=370$, $d_{CP}=3448$, $k=2666$;
或者, $r=370$, $d_{CP}=1936$, $k=3236$;
或者, $r=371$, $d_{CP}=1736$, $k=2672$;
或者, $r=384$, $d_{CP}=1488$, $k=3358$;
或者, $r=384$, $d_{CP}=3840$, $k=3360$;
30 或者, $r=385$, $d_{CP}=616$, $k=3366$;
或者, $r=385$, $d_{CP}=2968$, $k=3368$;
或者, $r=386$, $d_{CP}=2096$, $k=3376$;
或者, $r=387$, $d_{CP}=1224$, $k=3384$;
或者, $r=387$, $d_{CP}=3576$, $k=3386$;
35 或者, $r=388$, $d_{CP}=352$, $k=3392$;
或者, $r=388$, $d_{CP}=2704$, $k=3394$;
或者, $r=389$, $d_{CP}=1832$, $k=3402$;
或者, $r=391$, $d_{CP}=2440$, $k=2660$;
或者, $r=392$, $d_{CP}=1232$, $k=2666$;
40 或者, $r=392$, $d_{CP}=2240$, $k=3200$;
或者, $r=393$, $d_{CP}=24$, $k=2672$;
或者, $r=393$, $d_{CP}=3048$, $k=2674$;
或者, $r=393$, $d_{CP}=2040$, $k=3208$;
或者, $r=394$, $d_{CP}=1840$, $k=3216$;
45 或者, $r=395$, $d_{CP}=1640$, $k=3224$;
或者, $r=396$, $d_{CP}=1440$, $k=3232$;

- 或者, $r=396$, $d_{CP}=3960$, $k=3234$;
或者, $r=397$, $d_{CP}=1240$, $k=3240$;
或者, $r=397$, $d_{CP}=3760$, $k=3242$;
或者, $r=411$, $d_{CP}=960$, $k=3354$;
5 或者, $r=411$, $d_{CP}=3480$, $k=3356$;
或者, $r=412$, $d_{CP}=760$, $k=3362$;
或者, $r=412$, $d_{CP}=3280$, $k=3364$;
或者, $r=413$, $d_{CP}=2912$, $k=2662$;
或者, $r=413$, $d_{CP}=560$, $k=3370$;
10 或者, $r=413$, $d_{CP}=3080$, $k=3372$;
或者, $r=414$, $d_{CP}=2208$, $k=2668$;
或者, $r=414$, $d_{CP}=360$, $k=3378$;
或者, $r=414$, $d_{CP}=2880$, $k=3380$;
或者, $r=415$, $d_{CP}=1504$, $k=2674$;
15 或者, $r=415$, $d_{CP}=160$, $k=3386$;
或者, $r=415$, $d_{CP}=2680$, $k=3388$;
或者, $r=416$, $d_{CP}=2480$, $k=3396$;
或者, $r=418$, $d_{CP}=1072$, $k=3198$;
或者, $r=418$, $d_{CP}=3760$, $k=3200$;
20 或者, $r=419$, $d_{CP}=1544$, $k=3206$;
或者, $r=419$, $d_{CP}=4232$, $k=3208$;
或者, $r=420$, $d_{CP}=2016$, $k=3214$;
或者, $r=421$, $d_{CP}=2488$, $k=3222$;
或者, $r=422$, $d_{CP}=272$, $k=3228$;
25 或者, $r=422$, $d_{CP}=2960$, $k=3230$;
或者, $r=423$, $d_{CP}=744$, $k=3236$;
或者, $r=423$, $d_{CP}=3432$, $k=3238$;
或者, $r=434$, $d_{CP}=560$, $k=2656$;
或者, $r=434$, $d_{CP}=3920$, $k=2658$;
30 或者, $r=435$, $d_{CP}=360$, $k=2662$;
或者, $r=435$, $d_{CP}=3720$, $k=2664$;
或者, $r=436$, $d_{CP}=160$, $k=2668$;
或者, $r=436$, $d_{CP}=3520$, $k=2670$;
或者, $r=437$, $d_{CP}=3320$, $k=2676$;
35 或者, $r=438$, $d_{CP}=2448$, $k=3352$;
或者, $r=439$, $d_{CP}=232$, $k=3358$;
或者, $r=439$, $d_{CP}=2920$, $k=3360$;
或者, $r=440$, $d_{CP}=704$, $k=3366$;
或者, $r=440$, $d_{CP}=3392$, $k=3368$;
40 或者, $r=441$, $d_{CP}=1176$, $k=3374$;
或者, $r=441$, $d_{CP}=3864$, $k=3376$;
或者, $r=442$, $d_{CP}=1648$, $k=3382$;
或者, $r=442$, $d_{CP}=4336$, $k=3384$;
或者, $r=443$, $d_{CP}=2120$, $k=3390$;
45 或者, $r=444$, $d_{CP}=2424$, $k=3198$;
或者, $r=444$, $d_{CP}=2592$, $k=3398$;

- 或者, $r=445$, $d_{CP}=712$, $k=3204$;
或者, $r=445$, $d_{CP}=3568$, $k=3206$;
或者, $r=446$, $d_{CP}=1856$, $k=3212$;
或者, $r=447$, $d_{CP}=144$, $k=3218$;
5 或者, $r=447$, $d_{CP}=3000$, $k=3220$;
或者, $r=448$, $d_{CP}=1288$, $k=3226$;
或者, $r=448$, $d_{CP}=4144$, $k=3228$;
或者, $r=449$, $d_{CP}=2432$, $k=3234$;
或者, $r=456$, $d_{CP}=1032$, $k=2658$;
10 或者, $r=456$, $d_{CP}=4560$, $k=2660$;
或者, $r=457$, $d_{CP}=1336$, $k=2664$;
或者, $r=458$, $d_{CP}=1640$, $k=2670$;
或者, $r=459$, $d_{CP}=1944$, $k=2676$;
或者, $r=466$, $d_{CP}=1888$, $k=3356$;
15 或者, $r=466$, $d_{CP}=4744$, $k=3358$;
或者, $r=467$, $d_{CP}=176$, $k=3362$;
或者, $r=467$, $d_{CP}=3032$, $k=3364$;
或者, $r=468$, $d_{CP}=1320$, $k=3370$;
或者, $r=468$, $d_{CP}=4176$, $k=3372$;
20 或者, $r=469$, $d_{CP}=2464$, $k=3378$;
或者, $r=470$, $d_{CP}=752$, $k=3196$;
或者, $r=470$, $d_{CP}=3776$, $k=3198$;
或者, $r=470$, $d_{CP}=752$, $k=3384$;
或者, $r=470$, $d_{CP}=3608$, $k=3386$;
25 或者, $r=471$, $d_{CP}=2568$, $k=3204$;
或者, $r=471$, $d_{CP}=1896$, $k=3392$;
或者, $r=471$, $d_{CP}=4752$, $k=3394$;
或者, $r=472$, $d_{CP}=1360$, $k=3210$;
或者, $r=472$, $d_{CP}=4384$, $k=3212$;
30 或者, $r=472$, $d_{CP}=184$, $k=3398$;
或者, $r=472$, $d_{CP}=3040$, $k=3400$;
或者, $r=473$, $d_{CP}=152$, $k=3216$;
或者, $r=473$, $d_{CP}=3176$, $k=3218$;
或者, $r=474$, $d_{CP}=1968$, $k=3224$;
35 或者, $r=475$, $d_{CP}=760$, $k=3230$;
或者, $r=475$, $d_{CP}=3784$, $k=3232$;
或者, $r=476$, $d_{CP}=2576$, $k=3238$;
或者, $r=478$, $d_{CP}=1840$, $k=2660$;
或者, $r=479$, $d_{CP}=2648$, $k=2666$;
40 或者, $r=480$, $d_{CP}=3456$, $k=2672$;
或者, $r=481$, $d_{CP}=568$, $k=2676$;
或者, $r=481$, $d_{CP}=4264$, $k=2678$;
或者, $r=493$, $d_{CP}=184$, $k=3352$;
或者, $r=493$, $d_{CP}=3208$, $k=3354$;
45 或者, $r=494$, $d_{CP}=2000$, $k=3360$;
或者, $r=494$, $d_{CP}=5024$, $k=3362$;

- 或者, $r=495$, $d_{CP}=792$, $k=3366$;
 或者, $r=495$, $d_{CP}=3816$, $k=3368$;
 或者, $r=496$, $d_{CP}=1936$, $k=3196$;
 或者, $r=496$, $d_{CP}=2608$, $k=3374$;
 5 或者, $r=497$, $d_{CP}=1232$, $k=3202$;
 或者, $r=497$, $d_{CP}=4424$, $k=3204$;
 或者, $r=497$, $d_{CP}=1400$, $k=3380$;
 或者, $r=497$, $d_{CP}=4424$, $k=3382$;
 或者, $r=498$, $d_{CP}=528$, $k=3208$;
 10 或者, $r=498$, $d_{CP}=3720$, $k=3210$;
 或者, $r=498$, $d_{CP}=192$, $k=3386$;
 或者, $r=498$, $d_{CP}=3216$, $k=3388$;
 或者, $r=499$, $d_{CP}=1672$, $k=2656$;
 或者, $r=499$, $d_{CP}=3016$, $k=3216$;
 15 或者, $r=499$, $d_{CP}=2008$, $k=3394$;
 或者, $r=499$, $d_{CP}=5032$, $k=3396$;
 或者, $r=500$, $d_{CP}=2984$, $k=2662$;
 或者, $r=500$, $d_{CP}=2312$, $k=3222$;
 或者, $r=500$, $d_{CP}=800$, $k=3400$;
 20 或者, $r=500$, $d_{CP}=3824$, $k=3402$;
 或者, $r=501$, $d_{CP}=432$, $k=2666$;
 或者, $r=501$, $d_{CP}=4296$, $k=2668$;
 或者, $r=501$, $d_{CP}=1608$, $k=3228$;
 或者, $r=501$, $d_{CP}=4800$, $k=3230$;
 25 或者, $r=502$, $d_{CP}=1744$, $k=2672$;
 或者, $r=502$, $d_{CP}=904$, $k=3234$;
 或者, $r=502$, $d_{CP}=4096$, $k=3236$;
 或者, $r=503$, $d_{CP}=3056$, $k=2678$ 。
 在一些可能的实施方式中, $d_{scr} = 336 \times k$ 。
 30 在一些可能的实施方式中, 对所述第一数据进行 CRC 包括:
 对所述第一数据中每 r/p 行总共 $r \times q/p$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特,
 $d_{CRC} = 32 \times p$, r 能被 p 整除, p 为大于 1 的整数。
 或者,
 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32
 35 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的
 校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, $r_{F0} > r_{F1}$,
 $[a]$ 表示对 a 进行向下取整。
 在一些可能的实施方式中, $q \times r_{F0} = 41120$ 。
 在一些可能的实施方式中, $q=2056$ 。
 40 在一些可能的实施方式中, 对所述第一数据进行 CRC 包括: 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的
 每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后
 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。其中, $d_{CRC} = 32 \times p$, r 不能
 被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数 r_{F0} 大于整数 r_{F1} , $[a]$ 表示对 a 进行向下取整。
 在一些可能的实施方式中, $r=435$ 。
 45 在一些可能的实施方式中, $p=22$, $r_{F0}=20$, $r_{F1}=15$, 对所述第一数据进行 CRC 包括: 对所述第一数
 据中前 420 行中的每 20 行共 41120 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所

述第一数据中最后 15 行共 30840 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。

在一些可能的实施方式中, $d_{CP} = 3432$, $k = 2672$, 或者, $d_{CP} = 744$, $k = 2664$ 。

在一些可能的实施方式中, $r=546$, $d_{CP}=1008$, $k=3344$;

或者, $r=546$, $d_{CP}=3696$, $k=3352$;

5 或者, $r=547$, $d_{CP}=1640$, $k=3352$;

或者, $r=547$, $d_{CP}=4328$, $k=3360$;

或者, $r=548$, $d_{CP}=2272$, $k=3360$;

或者, $r=548$, $d_{CP}=4960$, $k=3368$;

或者, $r=549$, $d_{CP}=216$, $k=3360$;

10 或者, $r=549$, $d_{CP}=2904$, $k=3368$;

或者, $r=549$, $d_{CP}=5592$, $k=3376$;

或者, $r=550$, $d_{CP}=848$, $k=3368$;

或者, $r=550$, $d_{CP}=3536$, $k=3376$;

或者, $r=551$, $d_{CP}=1480$, $k=3376$;

15 或者, $r=551$, $d_{CP}=4168$, $k=3384$;

或者, $r=552$, $d_{CP}=2112$, $k=3384$;

或者, $r=552$, $d_{CP}=4800$, $k=3392$;

或者, $r=553$, $d_{CP}=56$, $k=3384$;

或者, $r=553$, $d_{CP}=2744$, $k=3392$;

20 或者, $r=553$, $d_{CP}=5432$, $k=3400$;

或者, $r=554$, $d_{CP}=688$, $k=3392$;

或者, $r=554$, $d_{CP}=3376$, $k=3400$;

或者, $r=555$, $d_{CP}=1320$, $k=3400$;

或者, $r=555$, $d_{CP}=4008$, $k=3408$;

25 或者, $r=556$, $d_{CP}=1952$, $k=3408$;

或者, $r=556$, $d_{CP}=4640$, $k=3416$;

或者, $r=557$, $d_{CP}=2584$, $k=3416$;

或者, $r=557$, $d_{CP}=5272$, $k=3424$;

或者, $r=558$, $d_{CP}=528$, $k=3416$;

30 或者, $r=558$, $d_{CP}=3216$, $k=3424$;

或者, $r=559$, $d_{CP}=1160$, $k=3424$;

或者, $r=559$, $d_{CP}=3848$, $k=3432$;

或者, $r=560$, $d_{CP}=1792$, $k=3432$;

或者, $r=560$, $d_{CP}=4480$, $k=3440$;

35 或者, $r=561$, $d_{CP}=2424$, $k=3440$;

或者, $r=561$, $d_{CP}=5112$, $k=3448$;

或者, $r=562$, $d_{CP}=368$, $k=3440$;

或者, $r=562$, $d_{CP}=3056$, $k=3448$;

或者, $r=562$, $d_{CP}=5744$, $k=3456$ 。

40 在一些可能的实施方式中, $r=520$, $d_{CP}=704$, $k=3184$;

或者, $r=520$, $d_{CP}=3392$, $k=3192$;

或者, $r=520$, $d_{CP}=6080$, $k=3200$;

或者, $r=521$, $d_{CP}=1336$, $k=3192$;

或者, $r=521$, $d_{CP}=4024$, $k=3200$;

45 或者, $r=522$, $d_{CP}=1968$, $k=3200$;

或者, $r=522$, $d_{CP}=4656$, $k=3208$;

- 或者, $r=523$, $d_{CP}=2600$, $k=3208$;
 或者, $r=523$, $d_{CP}=5288$, $k=3216$;
 或者, $r=524$, $d_{CP}=544$, $k=3208$;
 或者, $r=524$, $d_{CP}=3232$, $k=3216$;
 5 或者, $r=525$, $d_{CP}=1176$, $k=3216$;
 或者, $r=525$, $d_{CP}=3864$, $k=3224$;
 或者, $r=526$, $d_{CP}=1808$, $k=3224$;
 或者, $r=526$, $d_{CP}=4496$, $k=3232$;
 或者, $r=527$, $d_{CP}=2440$, $k=3232$;
 10 或者, $r=527$, $d_{CP}=5128$, $k=3240$;
 或者, $r=528$, $d_{CP}=384$, $k=3232$;
 或者, $r=528$, $d_{CP}=3072$, $k=3240$;
 或者, $r=528$, $d_{CP}=5760$, $k=3248$;
 或者, $r=529$, $d_{CP}=1016$, $k=3240$;
 15 或者, $r=529$, $d_{CP}=3704$, $k=3248$;
 或者, $r=530$, $d_{CP}=1648$, $k=3248$;
 或者, $r=530$, $d_{CP}=4336$, $k=3256$;
 或者, $r=531$, $d_{CP}=2280$, $k=3256$;
 或者, $r=531$, $d_{CP}=4968$, $k=3264$;
 20 或者, $r=532$, $d_{CP}=2912$, $k=3264$;
 或者, $r=532$, $d_{CP}=224$, $k=3256$;
 或者, $r=533$, $d_{CP}=856$, $k=3264$;
 或者, $r=533$, $d_{CP}=3544$, $k=3272$;
 或者, $r=534$, $d_{CP}=1488$, $k=3272$;
 25 或者, $r=534$, $d_{CP}=4176$, $k=3280$;
 或者, $r=535$, $d_{CP}=2120$, $k=3280$;
 或者, $r=535$, $d_{CP}=4808$, $k=3288$ 。

- 在一些可能的实施方式中, 对所述第二比特集合和所述第三比特集合进行所述 FEC 编码得到第四比特集合之后, 所述方法还包括: 对所述第四比特集合进行第一交织处理得到第五比特集合, 所述第五比特集合包括 m_0 个第二比特子集, 每个所述第二比特子集包括 256 个比特, 每个所述第二比特子集中的其中 128 个比特来自所述第二比特集合, 每个所述第二比特子集中的另外 128 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数。对 $L/2$ 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数。对所述双偏振符号流进行 DSP 成帧。

- 30 在一些可能的实施方式中, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

在一些可能的实施方式中, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

- 在一些可能的实施方式中, 对所述双偏振符号流进行 DSP 成帧包括: 将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

- 40 在一些可能的实施方式中, 从数据帧中获取第一数据包括: 从所述数据帧中获取 F_{Frame} 个第一数据, 以得到 $F_{Frame} \times r \times q$ 个比特, F_{Frame} 为大于 0 的整数。 $d_{scr} \times F_{Frame} = 336 \times k \times F_{DSP}$ 。其中, F_{DSP} 表示所述 DSP 超帧的数量, F_{DSP} 个 DSP 超帧由 F_{Frame} 个第一数据经过处理得到, F_{DSP} 为大于 0 的整数。

- 在一些可能的实施方式中, $F_{DSP} > 1$, 所述 F_{DSP} 个 DSP 超帧中第一 DSP 超帧中的连续多个符号为第一标识, 所述 F_{DSP} 个 DSP 超帧中除所述第一 DSP 超帧外的每个 DSP 超帧中的连续多个符号为第二标识。

- 45

第三方面, 本申请实施例提供了一种数据处理方法, 该数据处理方法包括如下步骤。对第一组 k 个

比特中的第一比特集合进行第一 PCS 处理得到第二比特集合, 所述第二比特集合包括 $m_0 \times 128$ 个比特, k 为大于 1 的整数, 所述 m_0 为大于 1 的整数。对第二比特集合和所述第一组 k 个比特中除所述第一比特集合外的第三比特集合进行第一 FEC 编码得到第四比特集合, 所述第四比特集合包括 m_0 个第一比特方阵, 每个所述第一比特方阵包括分布为 16 行 16 列的共 256 个比特。对第二组 k 个比特中的第五比特集合进行第二 PCS 处理得到第六比特集合。对第六比特集合和所述第二组 k 个比特中除所述第五比特集合外的第七比特集合进行第二 FEC 编码得到第八比特集合, 所述第八比特集合包括 m_0 个第二比特方阵, 每个所述第二比特方阵包括分布为 16 行 16 列的共 256 个比特。将 21 个所述第四比特集合和 21 个所述第八比特集合输入交织缓存。其中, 所述交织缓存包括分布为 84 行 m_1 列的共 $84 \times m_1$ 个缓存单元, $m_0 = m_1 \times 2$, 每个所述缓存单元用于缓存 16 行 16 列共 256 个比特。 $42 \times m_1$ 个所述第一比特方阵输入所述交织缓存的偶数行, $42 \times m_1$ 个所述第二比特方阵输入所述交织缓存的奇数行。所述交织缓存包括第一缓存子集和第二缓存子集, 所述第一缓存子集包括 $42 \times m_1$ 个所述缓存单元, 所述第二缓存子集包括 $42 \times m_1$ 个所述缓存单元。输入所述第一缓存子集的比特来自所述第二比特集合和所述第六比特集合, 输入所述第二缓存子集的比特来自所述第三比特集合、所述第七比特集合、第一 FEC 编码的校验比特和第二 FEC 编码的校验比特。对所述交织缓存中每 8 个比特进行符号映射和偏振划分得到 1 个双偏振 DP-16QAM 符号。所述 DP-16QAM 符号在目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个幅度位比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 DP-16QAM 符号在所述目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个符号位比特来自所述第二缓存子集的第二缓存单元中的一列。

在该实施方式中, 本申请第三方面提供的数据处理方法相对于上述第一方面介绍的数据处理方法来说, 提供了一种与上述第一方面中第二交织处理类似复杂度的第三交织处理, 该第三交织处理为第三方面采用交织缓存实现的操作, 第三交织处理与第二交织处理所用的缓存大小相近或相同。相当于第三交织处理替换了上述第一方面中的第一交织处理和第二交织处理, 达到了与第一方面类似的效果, 使得经过 PCS 处理得到的 0 和 1 不等概的比特可以映射到调制符号的幅度位比特, 实现了在保持星座点位置不变下改变星座点出现的概率, 使其非均匀分布, 提升了整体性能, 以满足未来更长传输距离需求, 并且, 第三方面提供的数据处理方法具有更低的复杂度。

在一些可能的实施方式中, 所述 8 个比特中的其中 4 个比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 8 个比特中的另外 4 个比特来自所述第二缓存子集的第二缓存单元中的一列。

在一些可能的实施方式中, 所述 m_0 个第一比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第一比特方阵和第 1 行第 m_2 列的第一比特方阵来自所述第二比特集合, 所述 m_0 个第二比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第二比特方阵和第 1 行第 m_2 列的第二比特方阵来自所述第六比特集合, $0 \leq m_2 < m_1/2$ 。

在一些可能的实施方式中, $m_0=16$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 2 列和第 3 列中的共 336 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 4 列、第 5 列、第 6 列和第 7 列中的共 336 个缓存单元。或者, $m_0=12$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列和第 2 列中的共 252 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 3 列、第 4 列和第 5 列中的共 252 个缓存单元。

在一些可能的实施方式中, $m_0=16$, $m_1=8$ 。所述 m_0 个第一比特方阵分布为 2 行 m_1 列, 第 0 行第 0 列的第一比特方阵、第 0 行第 1 列的第一比特方阵、第 0 行第 4 列的第一比特方阵、第 0 行第 5 列的第一比特方阵、第 1 行第 0 列的第一比特方阵、第 1 行第 1 列的第一比特方阵、第 1 行第 4 列的第一比特方阵和第 1 行第 5 列的第一比特方阵来自所述第二比特集合。所述 m_0 个第二比特方阵分布为 2 行 m_1 列, 第 0 行第 0 列的第二比特方阵、第 0 行第 1 列的第二比特方阵、第 0 行第 4 列的第二比特方阵、第 0 行第 5 列的第二比特方阵、第 1 行第 0 列的第二比特方阵、第 1 行第 1 列的第二比特方阵、第 1 行第 4 列的第二比特方阵和第 1 行第 5 列的第二比特方阵来自所述第六比特集合。

在一些可能的实施方式中, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 4 列和第 5 列中的共 336 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 2 列、第 3 列、第 6 列和第 7 列中的共 336 个缓存单元。

第四方面, 本申请实施例提供了一种数据处理装置, 该数据处理装置包括: PCS 单元、FEC 编码单元和第一交织处理单元。所述 PCS 单元用于: 对 k 个比特中的第一比特集合进行 PCS 处理得到第二比特集合, k 为大于 1 的整数。所述 FEC 单元用于: 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合, 所述第四比特集合包括 m_0 个第一比特子集, 每个所述第一比特子集包括 F_0 个比特, 所述 m_0 为大于 1 的整数, 所述 F_0 为大于 1 的偶数。所述第一交织处理单元用于: 对所述第四比特集合进行第一交织处理得到第五比特集合, 所述第五比特集合包括 m_0 个第二比特子集, 每个所述第二比特子集包括 F_0 个比特, 每个所述第二比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 每个所述第二比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

10 在一些可能的实施方式中, 每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列, 所述第二比特子集的每行 F_1 个比特中的其中 $F_1/2$ 个比特来自所述第二比特集合, 所述第二比特集合包括 $m_0 \times F_0/2$ 个比特, 所述第二比特子集的每行 F_1 个比特中的另外 $F_1/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

15 在一些可能的实施方式中, $F_0 = 256$, $F_1 = 16$, 所述第二比特子集中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特来自所述第二比特集合, $0 \leq i_2 < 16$, $0 \leq j_2 < 8$ 。或者, 所述第二比特子集中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自所述第二比特集合, $0 \leq i_2 < 16$, $0 \leq j_2 < 8$ 。其中, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

在一些可能的实施方式中, $F_0 = 256$, 所述第四比特集合中的 m_1 个所述第一比特子集总共 $m_1 \times 256$ 个比特来自所述第二比特集合, $m_0 = m_1 \times 2$ 。

20 在一些可能的实施方式中, $F_0 = 256$, 所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中, $0 \leq r_0 < 32$, $0 \leq r_1 < 32$ 。

在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第一比特子集 B_{0,m_2} 和第 1 行第 m_2 列的第一比特子集 B_{1,m_2} 来自所述第二比特集合, $0 \leq m_2 < m_1/2$ 。

25 在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 $\left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + c_0 \% 2 + (r_1 + 2) \% 4\right) \% (m_0 \times 8)$ 个比特来自所述第四比特集合中第 r_0 行的第 0 比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特, 其中, $0 \leq c_0 < (m_0 \times 4)$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特:

$$30 \quad c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特:

$$35 \quad c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times (m_0 \times 4)$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times (m_0 \times 4)$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合中第 r_0 行的第0比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < (m_0 \times 4)$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是2的整数倍, $c = c_0 \% (m_0 \times 4) \times 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是2的整数倍, $c = c_0 \% (m_0 \times 4) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是2的整数倍, $c = (c_0 \% 8) \times 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + 16 \times \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是2的整数倍, $c = (c_0 \% 8) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + 16 \times \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor$ 。

在一些可能的实施方式中, 所述数据处理装置还包括编码前交织单元。所述编码前交织单元用于: 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为2行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于1且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于1且小于 F_0 的偶数。所述FEC单元用于: 对所述第六比特集合进行FEC编码得到所述第四比特集合。

在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第0列至第5列的每个第三比特子集包括16行16列个比特, 第6列的每个第三比特子集包括16行15列个比特, 第0列的第三比特子集、第1列的第三比特子集、第4列的第三比特子集和第5列的第三比特子集中的比特来自所述第二比特集合, 第2列的第三比特子集、第3列的第三比特子集和第6列的第三比特子集中的比特来自所述第三比特集合。

在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为2行 m_1 列, $m_0=16$, $m_1=8$, 第0行第0列的第一比特子集 $B_{0,0}$ 、第0行第1列的第一比特子集 $B_{0,1}$ 、第0行第4列的第一比特子集 $B_{0,4}$ 、第0行第5列的第一比特子集 $B_{0,5}$ 、第1行第0列的第一比特子集 $B_{1,0}$ 、第1行第1列的第一比特子集 $B_{1,1}$ 、第1行第4列的第一比特子集 $B_{1,4}$ 和第1行第5列的第一比特子集 $B_{1,5}$ 来自所述第二比特集合。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的128个比特中第 $\left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 128$ 个比特来自所述第四比特集合中第 r_0 行的第0比特到第31比特或第64比特到第95比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < 64$, $[a]$ 表示对 a 进行向下取整, $b \% c$ 表示 b 对 c 取模。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是4的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是4的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自所述第四比特集合中第 r_0 行的128个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) \times 16$ 。

5 在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合中第 r_0 行的第 0 比特到第 31 比特或第 64 比特到第 95 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < 64$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

10 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 32) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

15 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 32) \times 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

20 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) \times 16$ 。

在一些可能的实施方式中, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

25 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = (c_0 \% 8) \times 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) \times 16$ 。

30 在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 3 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 3 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 4 列至第 5 列的每个第三比特子集中的 F_0 个比特来自所述第二比特集合, 第 6 列的每个第三比特子集中的 F_2 个比特来自所述第三比特集合。

在一些可能的实施方式中, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2 \right)$ 列的比特来自所述第二比特集合。或者, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $(j_1 \times 2)$ 列的比特来自所

述第二比特集合。其中, $0 \leq i_0 < 2$, $0 \leq j_0 < 4$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$, $\lfloor a \rfloor$ 表示对 a 进行向下取整, $b\%c$ 表示 b 对 c 取模。

5 在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$, 第 0 列至第 3 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 3 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 4 列至第 5 列的每个第一比特子集中的 F_0 个比特来自所述第二比特集合, 第 6 列至第 7 列的每个第一比特子集中的 F_0 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在一些可能的实施方式中, 所述第一交织处理单元具体用于: 对所述第四比特集合中第 4 列至第 7 列的第一比特子集进行第一交织处理得到对应的所述第五比特集合中第 4 列至第 7 列的第二比特子集。

10 在一些可能的实施方式中, 所述第五比特集合中第 r_1 行的 128 个比特中第 $64 + \left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 64$ 个比特来自所述第四比特集合中第 r_0 行的第 64 比特到第 95 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < 31$, $\lfloor a \rfloor$ 表示对 a 进行向下取整, $b\%c$ 表示 b 对 c 取模。

15 在一些可能的实施方式中, 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$;

或者,
20 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

25 其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$;

或者,
30 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$;

或者,
35 当 $0 \leq c_1 < 64$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $64 \leq c_0 < 128$, $64 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \times 2 + 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) \times 16$ 。

5 在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 0 行第 5 列的第三比特子集 $I_{0,5}$ 、第 1 行第 5 列的第三比特子集 $I_{1,5}$ 、第 0 行第 6 列的第三比特子集 $I_{0,6}$ 中第 0 列至第 7 列的比特和第 1 行第 6 列的第三比特子集 $I_{1,6}$ 中第 0 列至第 7 列的比特来自所述第二比特集合, 第 0 行第 6 列的第三比特子集 $I_{0,6}$ 中第 8 列至第 14 列的比特和第 1 行第 6 列的第三比特子集 $I_{1,6}$ 中第 8 列至第 14 列的比特来自所述第三比特集合。

在一些可能的实施方式中, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列的比特来自所述第二比特集合;

或者, 第 i_0 行第 j_0 列的第三比特子集中第 i_1 行第 $(j_1 \times 2)$ 列的比特来自所述第二比特集合;

15 其中, $0 \leq i_0 < 2$, $0 \leq j_0 < 5$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$, $[a]$ 表示对 a 进行向下取整, $b\%c$ 表示 b 对 c 取模。

20 在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$, 第 0 列至第 4 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合。第 0 行第 5 列的第一比特子集 $B_{0,5}$ 、第 1 行第 5 列的第一比特子集 $B_{1,5}$ 、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 0 行至第 7 行第 0 列至第 7 列的 8×8 个比特、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 8 行至第 15 行第 8 列至第 15 列的 8×8 个比特、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 0 行至第 7 行第 0 列至第 7 列的 8×8 个比特和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 8 行至第 15 行第 8 列至第 15 列的 8×8 个比特来自所述第二比特集合。第 0 行第 7 列的第一比特子集 $B_{0,7}$ 、第 1 行第 7 列的第一比特子集 $B_{1,7}$ 、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 0 行至第 7 行第 8 列至第 15 列的 8×8 个比特、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中第 8 行至第 15 行第 0 列至第 7 列的 8×8 个比特、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 0 行至第 7 行第 8 列至第 15 列的 8×8 个比特和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中第 8 行至第 15 行第 0 列至第 7 列的 8×8 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

在一些可能的实施方式中, 所述第一交织处理单元具体用于: 对所述第四比特集合中第 5 列至第 7 列的第一比特子集进行第一交织处理得到对应的所述第五比特集合中第 5 列至第 7 列的第二比特子集。

30 在一些可能的实施方式中, 当 $0 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 8$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第

$$80 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2)\% 4 \right) \% 48 \text{ 个比特来自所述第四比特集合中第 } r_0 \text{ 行的第 80 比特到第 103}$$

比特中的一个比特。当 $8 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 16$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 $80 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 +$

$$\tilde{c}_0 \% 2 + (r_1 + 2)\% 4 \right) \% 48 \text{ 个比特来自所述第四比特集合中第 } r_0 \text{ 行的第 80 比特到第 95 比特和第 104 比特}$$

到第 111 比特中的一个比特。其中, $0 \leq \tilde{c}_0 < 24$, $[a]$ 表示对 a 进行向下取整, $b\%c$ 表示 b 对 c 取模。

35 在一些可能的实施方式中, 当 $0 \leq c_1 < 80$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $80 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特;

$$c_1 = 80 + (c\% \theta + (r_1 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍, 且 48 能被 θ 整除, Δ 为整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(1 - \left\lfloor \frac{c_2}{24} \right\rfloor\right) \times 2, c_2 = (r_0 \% 16)^{\wedge} c_0;$$

或者,

- 5 当 $0 \leq c_1 < 80$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $80 \leq c_1 < 128$ 时, 所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特;

$$c_1 = 80 + (c\% \theta + (r_1 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍, 且 48 能被 θ 整除, Δ 为整数, β 是 4 的整数,

$$10 \quad c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(1 - \left\lfloor \frac{c_2}{24} \right\rfloor\right) \times 2, c_2 = (r_0 \% 16)^{\wedge} c_0.$$

- 在一些可能的实施方式中, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$ 。第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特。第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 5 列的每个第三比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第三比特子集中每行的另外 7 个比特来自所述第三比特集合, 第 6 列的每个第三比特子集中的 F_2 个比特来自所述第二比特集合。

- 15 在一些可能的实施方式中, 所述 m_0 个第一比特子集分布为 2 行 m_1 列, $m_0=16$, $m_1=8$ 。第 0 列至第 4 列的每个第一比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第一比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合。第 5 列的每个第一比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第一比特子集中每行的另外 7 个比特来自所述第三比特集合。第 6 列的每个第一比特子集中第 i_1 行第 $15 - i_1$ 列的比特来自所述 FEC 编码的校验比特, 第 6 列的每个第一比特子集中除了第 i_1 行第 $15 - i_1$ 列的比特之外的其他比特来自所述第二比特集合, $0 \leq i_1 \leq 15$ 。第 7 列的每个第一比特子集中的 F_0 个比特来自所述 FEC 编码的校验比特。

- 25 在一些可能的实施方式中, 所述第一交织处理单元具体用于: 对第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中第 C 列至第 15 列、第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中第 C 列至第 15 列、第 0 行第 6 列的第一比特子集 $B_{0,6}$ 、第 0 行第 7 列的第一比特子集 $B_{0,7}$ 、第 1 行第 6 列的第一比特子集 $B_{1,6}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 进行第一交织处理得到对应的第 0 行第 5 列的第二比特子集 $T_{0,5}$ 中第 C 列至第 15 列、第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中第 C 列至第 15 列、第 0 行第 6 列的第二比特子集 $T_{0,6}$ 、第 0 行第 7 列的第二比特子集 $T_{0,7}$ 、第 1 行第 6 列的第二比特子集 $T_{1,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 。其中, $0 \leq C \leq 15$ 。

- 30 在一些可能的实施方式中, $r_0 = r_1$ 。

在一些可能的实施方式中, 每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列, 所述第二比特子集中每行 F_1 个比特来自 2 个所述第一比特子集。

- 35 在一些可能的实施方式中, 每个所述第一比特子集中的 F_0 个比特分布为 F_1 行 F_1 列, 所述第二比特子集中每行 F_1 个比特的其中 $F_1/2$ 个比特来自其中一个第一比特子集的其中一行, 所述第二比特子集中每行 F_1 个比特的另外 $F_1/2$ 个比特来自另外一个第一比特子集的其中一行。

在一些可能的实施方式中, $F_0 = 256$, 所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中每列 32 个比特来自 2 个所述第一比特子集。

- 40 在一些可能的实施方式中, 每个所述第一比特子集中的 256 个比特分布为 16 行 16 列, 所述第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的其中一行, 所述第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的其中一行。

在一些可能的实施方式中,所述第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的第 $c_1\%16$ 列,所述第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的第 $c_1\%16$ 列,其中, $0 \leq c_1 < (m_0 \times 8)$ 。

在一些可能的实施方式中, $m_0=16$, 所述 16 个第二比特子集分布为 2 行 8 列;

- 5 第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特,
第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特,
第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特,
10 第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特,
第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,
15 第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,
第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,
第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。
20

在一些可能的实施方式中, $m_0=16$, 所述 16 个第二比特子集分布为 2 行 8 列;

- 第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特,
第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特,
25 第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,
第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,
30 第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特,
第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特,
第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,
35 第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

在一些可能的实施方式中, $m_0=16$, 所述 16 个第二比特子集分布为 2 行 8 列;

- 第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 0 行第 1 列的第二比特子集 $T_{0,1}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特,
40 第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 0 行第 3 列的第二比特子集 $T_{0,3}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特,
第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 0 行第 5 列的第二比特子集 $T_{0,5}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,
45 第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 0 行第 7 列的第二比特子集 $T_{0,7}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,

第 1 行第 0 列的第二比特子集 $T_{1,0}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特,

第 1 行第 2 列的第二比特子集 $T_{1,2}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ ，和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特，

5 第 1 行第 4 列的第二比特子集 $T_{1,4}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,

第 1 行第 6 列的第二比特子集 $T_{1,6}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

在一些可能的实施方式中，所述 16 个第二比特子集分布为 2 行 8 列：

10 第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特,

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 2 列的第一比特子集 $B_{1,2}$ 中共 512 个比特,

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 3 列的第一比特子集 $B_{0,3}$ 中共 512 个比特，

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 3 列的第一比特子集 $B_{1,3}$ 中共 512 个比特,

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 0 行第 4 列的第一比特子集 $B_{0,4}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,

第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 4 列的第一比特子集 $B_{1,4}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 0 行第 5 列的第一比特子集 $B_{0,5}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,

第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

在一些可能的实施方式中，所述 16 个第二比特子集分布为 2 行 8 列：

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特,

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 3 列的第一比特子集 $B_{0,3}$ 中共 512 个比特,

第0行第2列的第二比特子集 $T_{0,2}$ 和第1行第2列的第二比特子集 $T_{1,2}$ 中共512个比特来自第0行第4列的第一比特子集 $B_{0,4}$ 和第0行第6列的第一比特子集 $B_{0,6}$ 中共512个比特,

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 0 行第 5 列的第一比特子集 $B_{0,5}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,

35 第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 2 列的第一比特子集 $B_{1,2}$ 中共 512 个比特,

第0行第5列的第二比特子集 $T_{0,5}$ 和第1行第5列的第二比特子集 $T_{1,5}$ 中共512个比特来自第1行第1列的第一比特子集 $B_{1,1}$ 和第1行第3列的第一比特子集 $B_{1,3}$ 中共512个比特,

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 1 行第 4 列的第一比特子集 $B_{1,4}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,

第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

在一些可能的实施方式中, $m_0=16$, 所述 16 个第二比特子集分布为 2 行 8 列:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 0 行第 1 列的第二比特子集 $T_{0,1}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特，

第0行第2列的第二比特子集 $T_{0,2}$ 和第0行第3列的第二比特子集 $T_{0,3}$ 中共512个比特来自第0行第

1 列的第一比特子集 $B_{0,1}$ 和第0行第3列的第一比特子集 $B_{0,3}$ 中共512个比特,

第0行第4列的第二比特子集 $T_{0,4}$ 和第0行第5列的第二比特子集 $T_{0,5}$ 中共512个比特来自第0行第4列的第一比特子集 $B_{0,4}$ 和第0行第6列的第一比特子集 $B_{0,6}$ 中共512个比特,

5 第0行第6列的第二比特子集 $T_{0,6}$ 和第0行第7列的第二比特子集 $T_{0,7}$ 中共512个比特来自第0行第5列的第一比特子集 $B_{0,5}$ 和第0行第7列的第一比特子集 $B_{0,7}$ 中共512个比特,

第1行第0列的第二比特子集 $T_{1,0}$ 和第1行第1列的第二比特子集 $T_{1,1}$ 中共512个比特来自第1行第0列的第一比特子集 $B_{1,0}$ 和第1行第2列的第一比特子集 $B_{1,2}$ 中共512个比特,

第1行第2列的第二比特子集 $T_{1,2}$ 和第1行第3列的第二比特子集 $T_{1,3}$ 中共512个比特来自第1行第1列的第一比特子集 $B_{1,1}$ 和第1行第3列的第一比特子集 $B_{1,3}$ 中共512个比特,

10 第1行第4列的第二比特子集 $T_{1,4}$ 和第1行第5列的第二比特子集 $T_{1,5}$ 中共512个比特来自第1行第4列的第一比特子集 $B_{1,4}$ 和第1行第6列的第一比特子集 $B_{1,6}$ 中共512个比特,

第1行第6列的第二比特子集 $T_{1,6}$ 和第1行第7列的第二比特子集 $T_{1,7}$ 中共512个比特来自第1行第5列的第一比特子集 $B_{1,5}$ 和第1行第7列的第一比特子集 $B_{1,7}$ 中共512个比特。

在一些可能的实施方式中, $m_0=12$ 或 16。

15 在一些可能的实施方式中, 所述数据处理装置还包括第二交织处理单元、符号映射单元、偏振划分单元和 DSP 成帧单元。所述第二交织处理单元用于: 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 L/2 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数。所述符号映射单元和所述偏振划分单元用于: 对 L/2 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大

20 于 0 的整数。所述 DSP 成帧单元用于: 对所述双偏振符号流进行 DSP 成帧。

在一些可能的实施方式中, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

在一些可能的实施方式中, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

在一些可能的实施方式中, 所述 DSP 成帧单元具体用于: 将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

25

第五方面, 本申请实施例提供了一种数据处理装置, 该数据处理装置包括: 第一处理单元、第二处理单元、加扰单元、第三处理单元、PCS 单元和 FEC 编码单元。所述第一处理单元用于: 从数据帧中获取第一数据, 所述第一数据包括 r 行 q 列个比特, 所述 r 为大于 0 的整数, 所述 q 为大于 0 的整数。所述第二处理单元用于: 对所述第一数据进行循环冗余校验 CRC 和/或插入填充比特得到第二数据, 所述

30 第二数据包括进行 CRC 插入的 d_{CRC} 个校验比特和/或 d_{PAD} 个填充比特, d_{CRC} 为大于或等于 0 的整数, d_{PAD} 为大于或等于 0 的整数。加扰单元用于: 对所述第二数据进行加扰得到第三数据, 所述第三数据的比特数量是 $d_{scr} = r \times q + d_{CP}$, $d_{CP} = d_{CRC} + d_{PAD}$ 。所述第三处理单元用于: 获取所述第三数据中的 k 个比特, k 为大于 1 的整数, d_{scr} 是 k 的整数倍。所述 PCS 单元用于: 对 k 个比特中的第一比特集合进行 PCS 处理得到第二比特集合。所述 FEC 单元用于: 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合。

35 在一些可能的实施方式中, 所述数据处理装置还包括编码前交织单元, 所述编码前交织单元用于: 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数。所述 FEC 单元用于: 对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

在一些可能的实施方式中, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

45 在一些可能的实施方式中, d_{scr} 是 $4 \times k$ 的整数倍, $q=10280$ 。

在一些可能的实施方式中, 所述进行 CRC 和/或插入填充比特的冗余为 $(d_{CRC} + d_{PAD})/(r \times q)$,

$(d_{CRC} + d_{PAD})/(r \times q) \leq 0.001$ 。

- 在一些可能的实施方式中, $r=79$, $d_{CP}=328$, $k=3224$;
或者, $r=83$, $d_{CP}=32$, $k=3386$;
或者, $r=83$, $d_{CP}=536$, $k=3388$;
5 或者, $r=87$, $d_{CP}=72$, $k=2662$;
或者, $r=87$, $d_{CP}=744$, $k=2664$;
或者, $r=87$, $d_{CP}=3432$, $k=2672$;
或者, $r=104$, $d_{CP}=704$, $k=3184$;
或者, $r=104$, $d_{CP}=3392$, $k=3192$;
10 或者, $r=105$, $d_{CP}=504$, $k=3214$;
或者, $r=105$, $d_{CP}=1176$, $k=3216$;
或者, $r=109$, $d_{CP}=40$, $k=2668$;
或者, $r=109$, $d_{CP}=880$, $k=2670$;
或者, $r=110$, $d_{CP}=176$, $k=3366$;
15 或者, $r=110$, $d_{CP}=848$, $k=3368$;
或者, $r=110$, $d_{CP}=3536$, $k=3376$;
或者, $r=111$, $d_{CP}=648$, $k=3398$;
或者, $r=131$, $d_{CP}=8$, $k=2672$;
或者, $r=131$, $d_{CP}=1016$, $k=2674$;
20 或者, $r=131$, $d_{CP}=680$, $k=3208$;
或者, $r=132$, $d_{CP}=480$, $k=3232$;
或者, $r=132$, $d_{CP}=1320$, $k=3234$;
或者, $r=137$, $d_{CP}=320$, $k=3354$;
或者, $r=137$, $d_{CP}=1160$, $k=3356$;
25 或者, $r=138$, $d_{CP}=120$, $k=3378$;
或者, $r=138$, $d_{CP}=960$, $k=3380$;
或者, $r=152$, $d_{CP}=344$, $k=2658$;
或者, $r=152$, $d_{CP}=1520$, $k=2660$;
或者, $r=153$, $d_{CP}=648$, $k=2676$;
30 或者, $r=157$, $d_{CP}=856$, $k=3204$;
或者, $r=158$, $d_{CP}=656$, $k=3224$;
或者, $r=165$, $d_{CP}=264$, $k=3366$;
或者, $r=165$, $d_{CP}=1272$, $k=3368$;
或者, $r=166$, $d_{CP}=64$, $k=3386$;
35 或者, $r=166$, $d_{CP}=1072$, $k=3388$;
或者, $r=174$, $d_{CP}=144$, $k=2662$;
或者, $r=174$, $d_{CP}=1488$, $k=2664$;
或者, $r=175$, $d_{CP}=616$, $k=2678$;
或者, $r=183$, $d_{CP}=360$, $k=3200$;
40 或者, $r=183$, $d_{CP}=1536$, $k=3202$;
或者, $r=184$, $d_{CP}=664$, $k=3218$;
或者, $r=184$, $d_{CP}=1840$, $k=3220$;
或者, $r=185$, $d_{CP}=968$, $k=3236$;
或者, $r=192$, $d_{CP}=744$, $k=3358$;
45 或者, $r=192$, $d_{CP}=1920$, $k=3360$;
或者, $r=193$, $d_{CP}=1048$, $k=3376$;

- 或者, $r=194$, $d_{CP}=176$, $k=3392$;
或者, $r=194$, $d_{CP}=1352$, $k=3394$;
或者, $r=196$, $d_{CP}=616$, $k=2666$;
或者, $r=209$, $d_{CP}=536$, $k=3198$;
5 或者, $r=209$, $d_{CP}=1880$, $k=3200$;
或者, $r=210$, $d_{CP}=1008$, $k=3214$;
或者, $r=211$, $d_{CP}=136$, $k=3228$;
或者, $r=211$, $d_{CP}=1480$, $k=3230$;
或者, $r=217$, $d_{CP}=280$, $k=2656$;
10 或者, $r=217$, $d_{CP}=1960$, $k=2658$;
或者, $r=218$, $d_{CP}=80$, $k=2668$;
或者, $r=218$, $d_{CP}=1760$, $k=2670$;
或者, $r=219$, $d_{CP}=1224$, $k=3352$;
或者, $r=220$, $d_{CP}=352$, $k=3366$;
15 或者, $r=220$, $d_{CP}=1696$, $k=3368$;
或者, $r=221$, $d_{CP}=824$, $k=3382$;
或者, $r=221$, $d_{CP}=2168$, $k=3384$;
或者, $r=222$, $d_{CP}=1296$, $k=3398$;
或者, $r=235$, $d_{CP}=376$, $k=3196$;
20 或者, $r=235$, $d_{CP}=1888$, $k=3198$;
或者, $r=236$, $d_{CP}=680$, $k=3210$;
或者, $r=236$, $d_{CP}=2192$, $k=3212$;
或者, $r=237$, $d_{CP}=984$, $k=3224$;
或者, $r=238$, $d_{CP}=1288$, $k=3238$;
25 或者, $r=239$, $d_{CP}=920$, $k=2660$;
或者, $r=240$, $d_{CP}=1728$, $k=2672$;
或者, $r=247$, $d_{CP}=1000$, $k=3360$;
或者, $r=247$, $d_{CP}=2512$, $k=3362$;
或者, $r=248$, $d_{CP}=1304$, $k=3374$;
30 或者, $r=249$, $d_{CP}=96$, $k=3386$;
或者, $r=249$, $d_{CP}=1608$, $k=3388$;
或者, $r=250$, $d_{CP}=400$, $k=3400$;
或者, $r=250$, $d_{CP}=1912$, $k=3402$;
或者, $r=261$, $d_{CP}=216$, $k=2662$;
35 或者, $r=261$, $d_{CP}=2232$, $k=2664$;
或者, $r=261$, $d_{CP}=1560$, $k=3196$;
或者, $r=262$, $d_{CP}=16$, $k=2672$;
或者, $r=262$, $d_{CP}=2032$, $k=2674$;
或者, $r=262$, $d_{CP}=1360$, $k=3208$;
40 或者, $r=263$, $d_{CP}=1160$, $k=3220$;
或者, $r=264$, $d_{CP}=960$, $k=3232$;
或者, $r=264$, $d_{CP}=2640$, $k=3234$;
或者, $r=274$, $d_{CP}=640$, $k=3354$;
或者, $r=274$, $d_{CP}=2320$, $k=3356$;
45 或者, $r=275$, $d_{CP}=440$, $k=3366$;
或者, $r=275$, $d_{CP}=2120$, $k=3368$;

- 或者, $r=276$, $d_{CP}=240$, $k=3378$;
或者, $r=276$, $d_{CP}=1920$, $k=3380$;
或者, $r=277$, $d_{CP}=40$, $k=3390$;
或者, $r=277$, $d_{CP}=1720$, $k=3392$;
5 或者, $r=283$, $d_{CP}=2032$, $k=2666$;
或者, $r=284$, $d_{CP}=488$, $k=2674$;
或者, $r=284$, $d_{CP}=2672$, $k=2676$;
或者, $r=287$, $d_{CP}=896$, $k=3194$;
或者, $r=287$, $d_{CP}=2744$, $k=3196$;
10 或者, $r=288$, $d_{CP}=1704$, $k=3206$;
或者, $r=289$, $d_{CP}=664$, $k=3216$;
或者, $r=289$, $d_{CP}=2512$, $k=3218$;
或者, $r=290$, $d_{CP}=1472$, $k=3228$;
或者, $r=291$, $d_{CP}=432$, $k=3238$;
15 或者, $r=291$, $d_{CP}=2280$, $k=3240$;
或者, $r=302$, $d_{CP}=80$, $k=3360$;
或者, $r=302$, $d_{CP}=1928$, $k=3362$;
或者, $r=303$, $d_{CP}=888$, $k=3372$;
或者, $r=303$, $d_{CP}=2736$, $k=3374$;
20 或者, $r=304$, $d_{CP}=688$, $k=2658$;
或者, $r=304$, $d_{CP}=3040$, $k=2660$;
或者, $r=304$, $d_{CP}=1696$, $k=3384$;
或者, $r=305$, $d_{CP}=2168$, $k=2668$;
或者, $r=305$, $d_{CP}=656$, $k=3394$;
25 或者, $r=305$, $d_{CP}=2504$, $k=3396$;
或者, $r=306$, $d_{CP}=1296$, $k=2676$;
或者, $r=313$, $d_{CP}=1912$, $k=3194$;
或者, $r=314$, $d_{CP}=1712$, $k=3204$;
或者, $r=315$, $d_{CP}=1512$, $k=3214$;
30 或者, $r=316$, $d_{CP}=1312$, $k=3224$;
或者, $r=317$, $d_{CP}=1112$, $k=3234$;
或者, $r=317$, $d_{CP}=3128$, $k=3236$;
或者, $r=326$, $d_{CP}=320$, $k=2660$;
或者, $r=326$, $d_{CP}=2840$, $k=2662$;
35 或者, $r=327$, $d_{CP}=120$, $k=2668$;
或者, $r=327$, $d_{CP}=2640$, $k=2670$;
或者, $r=328$, $d_{CP}=2440$, $k=2678$;
或者, $r=329$, $d_{CP}=728$, $k=3356$;
或者, $r=329$, $d_{CP}=2744$, $k=3358$;
40 或者, $r=330$, $d_{CP}=528$, $k=3366$;
或者, $r=330$, $d_{CP}=2544$, $k=3368$;
或者, $r=331$, $d_{CP}=328$, $k=3376$;
或者, $r=331$, $d_{CP}=2344$, $k=3378$;
或者, $r=332$, $d_{CP}=128$, $k=3386$;
45 或者, $r=332$, $d_{CP}=2144$, $k=3388$;
或者, $r=333$, $d_{CP}=1944$, $k=3398$;

- 或者, $r=339$, $d_{CP}=744$, $k=3192$;
或者, $r=339$, $d_{CP}=2928$, $k=3194$;
或者, $r=340$, $d_{CP}=1384$, $k=3202$;
或者, $r=341$, $d_{CP}=2024$, $k=3212$;
5 或者, $r=342$, $d_{CP}=480$, $k=3220$;
或者, $r=342$, $d_{CP}=2664$, $k=3222$;
或者, $r=343$, $d_{CP}=1120$, $k=3230$;
或者, $r=343$, $d_{CP}=3304$, $k=3232$;
或者, $r=344$, $d_{CP}=1760$, $k=3240$;
10 或者, $r=348$, $d_{CP}=288$, $k=2662$;
或者, $r=348$, $d_{CP}=2976$, $k=2664$;
或者, $r=349$, $d_{CP}=760$, $k=2670$;
或者, $r=349$, $d_{CP}=3448$, $k=2672$;
或者, $r=350$, $d_{CP}=1232$, $k=2678$;
15 或者, $r=356$, $d_{CP}=704$, $k=3352$;
或者, $r=356$, $d_{CP}=2888$, $k=3354$;
或者, $r=357$, $d_{CP}=1344$, $k=3362$;
或者, $r=357$, $d_{CP}=3528$, $k=3364$;
或者, $r=358$, $d_{CP}=1984$, $k=3372$;
20 或者, $r=359$, $d_{CP}=440$, $k=3380$;
或者, $r=359$, $d_{CP}=2624$, $k=3382$;
或者, $r=360$, $d_{CP}=1080$, $k=3390$;
或者, $r=360$, $d_{CP}=3264$, $k=3392$;
或者, $r=361$, $d_{CP}=1720$, $k=3400$;
25 或者, $r=365$, $d_{CP}=1592$, $k=3192$;
或者, $r=366$, $d_{CP}=720$, $k=3200$;
或者, $r=366$, $d_{CP}=3072$, $k=3202$;
或者, $r=367$, $d_{CP}=2200$, $k=3210$;
或者, $r=368$, $d_{CP}=1328$, $k=3218$;
30 或者, $r=368$, $d_{CP}=3680$, $k=3220$;
或者, $r=369$, $d_{CP}=2304$, $k=2658$;
或者, $r=369$, $d_{CP}=456$, $k=3226$;
或者, $r=369$, $d_{CP}=2808$, $k=3228$;
或者, $r=370$, $d_{CP}=592$, $k=2664$;
35 或者, $r=370$, $d_{CP}=3448$, $k=2666$;
或者, $r=370$, $d_{CP}=1936$, $k=3236$;
或者, $r=371$, $d_{CP}=1736$, $k=2672$;
或者, $r=384$, $d_{CP}=1488$, $k=3358$;
或者, $r=384$, $d_{CP}=3840$, $k=3360$;
40 或者, $r=385$, $d_{CP}=616$, $k=3366$;
或者, $r=385$, $d_{CP}=2968$, $k=3368$;
或者, $r=386$, $d_{CP}=2096$, $k=3376$;
或者, $r=387$, $d_{CP}=1224$, $k=3384$;
或者, $r=387$, $d_{CP}=3576$, $k=3386$;
45 或者, $r=388$, $d_{CP}=352$, $k=3392$;
或者, $r=388$, $d_{CP}=2704$, $k=3394$;

- 或者, $r=389$, $d_{CP}=1832$, $k=3402$;
或者, $r=391$, $d_{CP}=2440$, $k=2660$;
或者, $r=392$, $d_{CP}=1232$, $k=2666$;
或者, $r=392$, $d_{CP}=2240$, $k=3200$;
5 或者, $r=393$, $d_{CP}=24$, $k=2672$;
或者, $r=393$, $d_{CP}=3048$, $k=2674$;
或者, $r=393$, $d_{CP}=2040$, $k=3208$;
或者, $r=394$, $d_{CP}=1840$, $k=3216$;
或者, $r=395$, $d_{CP}=1640$, $k=3224$;
10 或者, $r=396$, $d_{CP}=1440$, $k=3232$;
或者, $r=396$, $d_{CP}=3960$, $k=3234$;
或者, $r=397$, $d_{CP}=1240$, $k=3240$;
或者, $r=397$, $d_{CP}=3760$, $k=3242$;
或者, $r=411$, $d_{CP}=960$, $k=3354$;
15 或者, $r=411$, $d_{CP}=3480$, $k=3356$;
或者, $r=412$, $d_{CP}=760$, $k=3362$;
或者, $r=412$, $d_{CP}=3280$, $k=3364$;
或者, $r=413$, $d_{CP}=2912$, $k=2662$;
或者, $r=413$, $d_{CP}=560$, $k=3370$;
20 或者, $r=413$, $d_{CP}=3080$, $k=3372$;
或者, $r=414$, $d_{CP}=2208$, $k=2668$;
或者, $r=414$, $d_{CP}=360$, $k=3378$;
或者, $r=414$, $d_{CP}=2880$, $k=3380$;
或者, $r=415$, $d_{CP}=1504$, $k=2674$;
25 或者, $r=415$, $d_{CP}=160$, $k=3386$;
或者, $r=415$, $d_{CP}=2680$, $k=3388$;
或者, $r=416$, $d_{CP}=2480$, $k=3396$;
或者, $r=418$, $d_{CP}=1072$, $k=3198$;
或者, $r=418$, $d_{CP}=3760$, $k=3200$;
30 或者, $r=419$, $d_{CP}=1544$, $k=3206$;
或者, $r=419$, $d_{CP}=4232$, $k=3208$;
或者, $r=420$, $d_{CP}=2016$, $k=3214$;
或者, $r=421$, $d_{CP}=2488$, $k=3222$;
或者, $r=422$, $d_{CP}=272$, $k=3228$;
35 或者, $r=422$, $d_{CP}=2960$, $k=3230$;
或者, $r=423$, $d_{CP}=744$, $k=3236$;
或者, $r=423$, $d_{CP}=3432$, $k=3238$;
或者, $r=434$, $d_{CP}=560$, $k=2656$;
或者, $r=434$, $d_{CP}=3920$, $k=2658$;
40 或者, $r=435$, $d_{CP}=360$, $k=2662$;
或者, $r=435$, $d_{CP}=3720$, $k=2664$;
或者, $r=436$, $d_{CP}=160$, $k=2668$;
或者, $r=436$, $d_{CP}=3520$, $k=2670$;
或者, $r=437$, $d_{CP}=3320$, $k=2676$;
45 或者, $r=438$, $d_{CP}=2448$, $k=3352$;
或者, $r=439$, $d_{CP}=232$, $k=3358$;

- 或者, $r=439$, $d_{CP}=2920$, $k=3360$;
或者, $r=440$, $d_{CP}=704$, $k=3366$;
或者, $r=440$, $d_{CP}=3392$, $k=3368$;
或者, $r=441$, $d_{CP}=1176$, $k=3374$;
5 或者, $r=441$, $d_{CP}=3864$, $k=3376$;
或者, $r=442$, $d_{CP}=1648$, $k=3382$;
或者, $r=442$, $d_{CP}=4336$, $k=3384$;
或者, $r=443$, $d_{CP}=2120$, $k=3390$;
或者, $r=444$, $d_{CP}=2424$, $k=3198$;
10 或者, $r=444$, $d_{CP}=2592$, $k=3398$;
或者, $r=445$, $d_{CP}=712$, $k=3204$;
或者, $r=445$, $d_{CP}=3568$, $k=3206$;
或者, $r=446$, $d_{CP}=1856$, $k=3212$;
或者, $r=447$, $d_{CP}=144$, $k=3218$;
15 或者, $r=447$, $d_{CP}=3000$, $k=3220$;
或者, $r=448$, $d_{CP}=1288$, $k=3226$;
或者, $r=448$, $d_{CP}=4144$, $k=3228$;
或者, $r=449$, $d_{CP}=2432$, $k=3234$;
或者, $r=456$, $d_{CP}=1032$, $k=2658$;
20 或者, $r=456$, $d_{CP}=4560$, $k=2660$;
或者, $r=457$, $d_{CP}=1336$, $k=2664$;
或者, $r=458$, $d_{CP}=1640$, $k=2670$;
或者, $r=459$, $d_{CP}=1944$, $k=2676$;
或者, $r=466$, $d_{CP}=1888$, $k=3356$;
25 或者, $r=466$, $d_{CP}=4744$, $k=3358$;
或者, $r=467$, $d_{CP}=176$, $k=3362$;
或者, $r=467$, $d_{CP}=3032$, $k=3364$;
或者, $r=468$, $d_{CP}=1320$, $k=3370$;
或者, $r=468$, $d_{CP}=4176$, $k=3372$;
30 或者, $r=469$, $d_{CP}=2464$, $k=3378$;
或者, $r=470$, $d_{CP}=752$, $k=3196$;
或者, $r=470$, $d_{CP}=3776$, $k=3198$;
或者, $r=470$, $d_{CP}=752$, $k=3384$;
或者, $r=470$, $d_{CP}=3608$, $k=3386$;
35 或者, $r=471$, $d_{CP}=2568$, $k=3204$;
或者, $r=471$, $d_{CP}=1896$, $k=3392$;
或者, $r=471$, $d_{CP}=4752$, $k=3394$;
或者, $r=472$, $d_{CP}=1360$, $k=3210$;
或者, $r=472$, $d_{CP}=4384$, $k=3212$;
40 或者, $r=472$, $d_{CP}=184$, $k=3398$;
或者, $r=472$, $d_{CP}=3040$, $k=3400$;
或者, $r=473$, $d_{CP}=152$, $k=3216$;
或者, $r=473$, $d_{CP}=3176$, $k=3218$;
或者, $r=474$, $d_{CP}=1968$, $k=3224$;
45 或者, $r=475$, $d_{CP}=760$, $k=3230$;
或者, $r=475$, $d_{CP}=3784$, $k=3232$;

- 或者, $r=476$, $d_{CP}=2576$, $k=3238$;
 或者, $r=478$, $d_{CP}=1840$, $k=2660$;
 或者, $r=479$, $d_{CP}=2648$, $k=2666$;
 或者, $r=480$, $d_{CP}=3456$, $k=2672$;
 5 或者, $r=481$, $d_{CP}=568$, $k=2676$;
 或者, $r=481$, $d_{CP}=4264$, $k=2678$;
 或者, $r=493$, $d_{CP}=184$, $k=3352$;
 或者, $r=493$, $d_{CP}=3208$, $k=3354$;
 或者, $r=494$, $d_{CP}=2000$, $k=3360$;
 10 或者, $r=494$, $d_{CP}=5024$, $k=3362$;
 或者, $r=495$, $d_{CP}=792$, $k=3366$;
 或者, $r=495$, $d_{CP}=3816$, $k=3368$;
 或者, $r=496$, $d_{CP}=1936$, $k=3196$;
 或者, $r=496$, $d_{CP}=2608$, $k=3374$;
 15 或者, $r=497$, $d_{CP}=1232$, $k=3202$;
 或者, $r=497$, $d_{CP}=4424$, $k=3204$;
 或者, $r=497$, $d_{CP}=1400$, $k=3380$;
 或者, $r=497$, $d_{CP}=4424$, $k=3382$;
 或者, $r=498$, $d_{CP}=528$, $k=3208$;
 20 或者, $r=498$, $d_{CP}=3720$, $k=3210$;
 或者, $r=498$, $d_{CP}=192$, $k=3386$;
 或者, $r=498$, $d_{CP}=3216$, $k=3388$;
 或者, $r=499$, $d_{CP}=1672$, $k=2656$;
 或者, $r=499$, $d_{CP}=3016$, $k=3216$;
 25 或者, $r=499$, $d_{CP}=2008$, $k=3394$;
 或者, $r=499$, $d_{CP}=5032$, $k=3396$;
 或者, $r=500$, $d_{CP}=2984$, $k=2662$;
 或者, $r=500$, $d_{CP}=2312$, $k=3222$;
 或者, $r=500$, $d_{CP}=800$, $k=3400$;
 30 或者, $r=500$, $d_{CP}=3824$, $k=3402$;
 或者, $r=501$, $d_{CP}=432$, $k=2666$;
 或者, $r=501$, $d_{CP}=4296$, $k=2668$;
 或者, $r=501$, $d_{CP}=1608$, $k=3228$;
 或者, $r=501$, $d_{CP}=4800$, $k=3230$;
 35 或者, $r=502$, $d_{CP}=1744$, $k=2672$;
 或者, $r=502$, $d_{CP}=904$, $k=3234$;
 或者, $r=502$, $d_{CP}=4096$, $k=3236$;
 或者, $r=503$, $d_{CP}=3056$, $k=2678$ 。

在一些可能的实施方式中, $d_{scr} = 336 \times k$ 。

- 40 在一些可能的实施方式中, 所述第二数据处理单元具体用于: 对所述第一数据中每 r/p 行总共 $r \times q/p$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, $d_{CRC} = 32 \times p$, r 能被 p 整除, p 为大于 1 的整数。或者, 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$,
 45 $r_{F0} > r_{F1}$, $[a]$ 表示对 a 进行向下取整。

在一些可能的实施方式中, $q \times r_{F0} = 41120$ 。

在一些可能的实施方式中, $q=2056$ 。

在一些可能的实施方式中, 所述第二数据处理单元具体用于: 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数 r_{F0} 大于整数 r_{F1} , $[a]$ 表示对 a 进行向下取整。

在一些可能的实施方式中, $r=435$ 。

在一些可能的实施方式中, $p=22$, $r_{F0}=20$, $r_{F1}=15$, 所述第二数据处理单元具体用于: 对所述第一数据中前 420 行中的每 20 行共 41120 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述

第一数据中最后 15 行共 30840 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。

在一些可能的实施方式中, $d_{CP} = 3432$, $k = 2672$, 或者, $d_{CP} = 744$, $k = 2664$ 。

在一些可能的实施方式中, $r=546$, $d_{CP}=1008$, $k=3344$;

或者, $r=546$, $d_{CP}=3696$, $k=3352$;

或者, $r=547$, $d_{CP}=1640$, $k=3352$;

15 或者, $r=547$, $d_{CP}=4328$, $k=3360$;

或者, $r=548$, $d_{CP}=2272$, $k=3360$;

或者, $r=548$, $d_{CP}=4960$, $k=3368$;

或者, $r=549$, $d_{CP}=216$, $k=3360$;

或者, $r=549$, $d_{CP}=2904$, $k=3368$;

20 或者, $r=549$, $d_{CP}=5592$, $k=3376$;

或者, $r=550$, $d_{CP}=848$, $k=3368$;

或者, $r=550$, $d_{CP}=3536$, $k=3376$;

或者, $r=551$, $d_{CP}=1480$, $k=3376$;

或者, $r=551$, $d_{CP}=4168$, $k=3384$;

25 或者, $r=552$, $d_{CP}=2112$, $k=3384$;

或者, $r=552$, $d_{CP}=4800$, $k=3392$;

或者, $r=553$, $d_{CP}=56$, $k=3384$;

或者, $r=553$, $d_{CP}=2744$, $k=3392$;

或者, $r=553$, $d_{CP}=5432$, $k=3400$;

30 或者, $r=554$, $d_{CP}=688$, $k=3392$;

或者, $r=554$, $d_{CP}=3376$, $k=3400$;

或者, $r=555$, $d_{CP}=1320$, $k=3400$;

或者, $r=555$, $d_{CP}=4008$, $k=3408$;

或者, $r=556$, $d_{CP}=1952$, $k=3408$;

35 或者, $r=556$, $d_{CP}=4640$, $k=3416$;

或者, $r=557$, $d_{CP}=2584$, $k=3416$;

或者, $r=557$, $d_{CP}=5272$, $k=3424$;

或者, $r=558$, $d_{CP}=528$, $k=3416$;

或者, $r=558$, $d_{CP}=3216$, $k=3424$;

40 或者, $r=559$, $d_{CP}=1160$, $k=3424$;

或者, $r=559$, $d_{CP}=3848$, $k=3432$;

或者, $r=560$, $d_{CP}=1792$, $k=3432$;

或者, $r=560$, $d_{CP}=4480$, $k=3440$;

或者, $r=561$, $d_{CP}=2424$, $k=3440$;

45 或者, $r=561$, $d_{CP}=5112$, $k=3448$;

或者, $r=562$, $d_{CP}=368$, $k=3440$;

- 或者, $r=562$, $d_{CP}=3056$, $k=3448$;
或者, $r=562$, $d_{CP}=5744$, $k=3456$ 。
在一些可能的实施方式中, $r=520$, $d_{CP}=704$, $k=3184$;
或者, $r=520$, $d_{CP}=3392$, $k=3192$;
5 或者, $r=520$, $d_{CP}=6080$, $k=3200$;
或者, $r=521$, $d_{CP}=1336$, $k=3192$;
或者, $r=521$, $d_{CP}=4024$, $k=3200$;
或者, $r=522$, $d_{CP}=1968$, $k=3200$;
或者, $r=522$, $d_{CP}=4656$, $k=3208$;
10 或者, $r=523$, $d_{CP}=2600$, $k=3208$;
或者, $r=523$, $d_{CP}=5288$, $k=3216$;
或者, $r=524$, $d_{CP}=544$, $k=3208$;
或者, $r=524$, $d_{CP}=3232$, $k=3216$;
或者, $r=525$, $d_{CP}=1176$, $k=3216$;
15 或者, $r=525$, $d_{CP}=3864$, $k=3224$;
或者, $r=526$, $d_{CP}=1808$, $k=3224$;
或者, $r=526$, $d_{CP}=4496$, $k=3232$;
或者, $r=527$, $d_{CP}=2440$, $k=3232$;
或者, $r=527$, $d_{CP}=5128$, $k=3240$;
20 或者, $r=528$, $d_{CP}=384$, $k=3232$;
或者, $r=528$, $d_{CP}=3072$, $k=3240$;
或者, $r=528$, $d_{CP}=5760$, $k=3248$;
或者, $r=529$, $d_{CP}=1016$, $k=3240$;
或者, $r=529$, $d_{CP}=3704$, $k=3248$;
25 或者, $r=530$, $d_{CP}=1648$, $k=3248$;
或者, $r=530$, $d_{CP}=4336$, $k=3256$;
或者, $r=531$, $d_{CP}=2280$, $k=3256$;
或者, $r=531$, $d_{CP}=4968$, $k=3264$;
或者, $r=532$, $d_{CP}=2912$, $k=3264$;
30 或者, $r=532$, $d_{CP}=224$, $k=3256$;
或者, $r=533$, $d_{CP}=856$, $k=3264$;
或者, $r=533$, $d_{CP}=3544$, $k=3272$;
或者, $r=534$, $d_{CP}=1488$, $k=3272$;
或者, $r=534$, $d_{CP}=4176$, $k=3280$;
35 或者, $r=535$, $d_{CP}=2120$, $k=3280$;
或者, $r=535$, $d_{CP}=4808$, $k=3288$ 。

在一些可能的实施方式中, 所述数据处理装置还包括第一交织单元、第二交织单元、符号映射单元、偏振划分单元和 DSP 成帧单元。所述第一交织单元用于: 对所述第四比特集合进行第一交织处理得到第五比特集合, 所述第五比特集合包括 m_0 个第二比特子集, 每个所述第二比特子集包括 256 个比特, 每个
40 所述第二比特子集中的其中 128 个比特来自所述第二比特集合, 每个所述第二比特子集中的另外 128 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。所述第二交织单元用于: 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数。所述符号映射单元和所述偏振划分单元用于: 对 $L/2$ 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特
45 映射为 1 个双偏振符号, 所述 t 为大于 0 的整数。所述 DSP 成帧单元用于: 对所述双偏振符号流进行 DSP 成帧。

在一些可能的实施方式中,所述双偏振符号中的幅度位比特来自所述第二比特集合。

在一些可能的实施方式中, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

在一些可能的实施方式中,所述 DSP 成帧单元具体用于:将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧,所述超帧包含 175104 个双偏振符号。

5 在一些可能的实施方式中,所述第一处理单元具体用于:从所述数据帧中获取 F_{Frame} 个第一数据,以得到 $F_{\text{Frame}} \times r \times q$ 个比特, F_{Frame} 为大于 0 的整数。 $d_{\text{scr}} \times F_{\text{Frame}} = 336 \times k \times F_{\text{DSP}}$, 其中, F_{DSP} 表示所述 DSP 超帧的数量, F_{DSP} 个 DSP 超帧由 F_{Frame} 个第一数据经过处理得到, F_{DSP} 为大于 0 的整数。

在一些可能的实施方式中, $F_{\text{DSP}} > 1$, 所述 F_{DSP} 个 DSP 超帧中第一 DSP 超帧中的连续多个符号为第一标识, 所述 F_{DSP} 个 DSP 超帧中除所述第一 DSP 超帧外的每个 DSP 超帧中的连续多个符号为第二标识。

10

第六方面,本申请实施例提供了一种数据处理装置,该数据处理装置包括:第一 PCS 单元、第一 FEC 编码单元、第二 PCS 单元、第二 FEC 单元、交织处理单元、符号映射单元和偏振划分单元。所述第一 PCS 单元用于:对第一组 k 个比特中的第一比特集合进行第一 PCS 处理得到第二比特集合,所述第二比特集合包括 $m_0 \times 128$ 个比特, k 为大于 1 的整数,所述 m_0 为大于 1 的整数。所述第一 FEC 编码单元用于:对第二比特集合和所述第一组 k 个比特中除所述第一比特集合外的第三比特集合进行第一前向纠错 FEC 编码得到第四比特集合,所述第四比特集合包括 m_0 个第一比特方阵,每个所述第一比特方阵包括分布为 16 行 16 列的共 256 个比特。所述第二 PCS 单元用于:对第二组 k 个比特中的第五比特集合进行第二 PCS 处理得到第六比特集合。所述第二 FEC 编码单元用于:对第六比特集合和所述第二组 k 个比特中除所述第五比特集合外的第七比特集合进行第二 FEC 编码得到第八比特集合,所述第八比特集合包括 m_0 个第二比特方阵,每个所述第二比特方阵包括分布为 16 行 16 列的共 256 个比特。所述交织处理单元用于:将 21 个所述第四比特集合和 21 个所述第八比特集合输入交织缓存,其中,所述交织缓存包括分布为 84 行 m_1 列的共 $84 \times m_1$ 个缓存单元, $m_0 = m_1 \times 2$, 每个所述缓存单元用于缓存 16 行 16 列共 256 个比特, $42 \times m_1$ 个所述第一比特方阵输入所述交织缓存的偶数行, $42 \times m_1$ 个所述第二比特方阵输入所述交织缓存的奇数行,所述交织缓存包括第一缓存子集和第二缓存子集,所述第一缓存子集包括 $42 \times m_1$ 个所述缓存单元,所述第二缓存子集包括 $42 \times m_1$ 个所述缓存单元,输入所述第一缓存子集的比特来自所述第二比特集合和所述第六比特集合,输入所述第二缓存子集的比特来自所述第三比特集合、所述第七比特集合、第一 FEC 编码的校验比特和第二 FEC 编码的校验比特。所述符号映射单元和所述偏振划分单元用于:对所述交织缓存中每 8 个比特进行符号映射和偏振划分得到 1 个双偏振 DP-16QAM 符号,所述 DP-16QAM 符号在目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个幅度位比特来自所述第一缓存子集的第一缓存单元中的一列,所述 DP-16QAM 符号在所述目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个符号位比特来自所述第二缓存子集的第二缓存单元中的一列。

在一些可能的实施方式中,所述 8 个比特中的其中 4 个比特来自所述第一缓存子集的第一缓存单元中的一列,所述 8 个比特中的另外 4 个比特来自所述第二缓存子集的第二缓存单元中的一列。

15 在一些可能的实施方式中,所述 m_0 个第一比特方阵分布为 2 行 m_1 列,其中,第 0 行第 m_2 列的第一比特方阵和第 1 行第 m_2 列的第一比特方阵来自所述第二比特集合,所述 m_0 个第二比特方阵分布为 2 行 m_1 列,其中,第 0 行第 m_2 列的第二比特方阵和第 1 行第 m_2 列的第二比特方阵来自所述第六比特集合, $0 \leq m_2 < m_1/2$ 。

20 在一些可能的实施方式中, $m_0=16$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 2 列和第 3 列中的共 336 个缓存单元,所述第二缓存子集包括所述交织缓存中第 4 列、第 5 列、第 6 列和第 7 列中的共 336 个缓存单元。或者, $m_0=12$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列和第 2 列中的共 252 个缓存单元,所述第二缓存子集包括所述交织缓存中第 3 列、第 4 列和第 5 列中的共 252 个缓存单元。

25 在一些可能的实施方式中, $m_0=16$, $m_1=8$ 。所述 m_0 个第一比特方阵分布为 2 行 m_1 列,第 0 行第 0 列的第一比特方阵、第 0 行第 1 列的第一比特方阵、第 0 行第 4 列的第一比特方阵、第 0 行第 5 列的第一比特方阵、第 1 行第 0 列的第一比特方阵、第 1 行第 1 列的第一比特方阵、第 1 行第 4 列的第一比特方阵和第 1 行第 5 列的第一比特方阵来自所述第二比特集合。所述 m_0 个第二比特方阵分布为 2 行 m_1 列,

45

第 0 行第 0 列的第二比特方阵、第 0 行第 1 列的第二比特方阵、第 0 行第 4 列的第二比特方阵、第 0 行第 5 列的第二比特方阵、第 1 行第 0 列的第二比特方阵、第 1 行第 1 列的第二比特方阵、第 1 行第 4 列的第二比特方阵和第 1 行第 5 列的第二比特方阵来自所述第六比特集合。

5 在一些可能的实施方式中，所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 4 列和第 5 列中的共 336 个缓存单元，所述第二缓存子集包括所述交织缓存中第 2 列、第 3 列、第 6 列和第 7 列中的共 336 个缓存单元。

附图说明

- 图 1 为本申请实施例应用的一种通信系统示意图；
- 10 图 2 为一种数据帧的结构示意图；
- 图 3 为本申请实施例中第一数据处理的一个实施例示意图；
- 图 4 为本申请实施例中第二数据处理的一个实施例示意图；
- 图 5 为本申请实施例中 PCS 和 OFEC 编码的一个实施例示意图；
- 图 6 为本申请实施例中第四比特集合的一个示意图；
- 15 图 7 为本申请实施例中第一比特子集的一个示意图；
- 图 8 为本申请实施例中第五比特集合的一个示意图；
- 图 9 (a) 为本申请实施例中第二比特子集的一个示意图；
- 图 9 (b) 为本申请实施例中第二比特子集的另一个示意图；
- 图 10 为本申请实施例中第一交织处理的第一种实施方式示意图；
- 20 图 11 (a) 为本申请实施例中多个第一比特子集的一种分布示意图；
- 图 11 (b) 为本申请实施例中多个第一比特子集的另一种分布示意图；
- 图 11 (c) 为本申请实施例中多个第一比特子集的另一种分布示意图；
- 图 11 (d) 为本申请实施例中多个第一比特子集的另一种分布示意图；
- 图 11 (e) 为本申请实施例中多个第一比特子集的另一种分布示意图；
- 25 图 12 为本申请实施例中第一交织处理的第二种实施方式示意图；
- 图 13 为本申请实施例中第一交织处理的第三种实施方式示意图；
- 图 14 为本申请实施例中方阵内交织的一个示意图；
- 图 15 为本申请实施例中方阵间交织的一个示意图；
- 图 16 为本申请实施例中方阵内交织的另一个示意图；
- 30 图 17 (a) 为本申请实施例中对数据帧进行 CRC 的一种实施方式示意图；
- 图 17 (b) 为本申请实施例中对数据帧进行 CRC 的另一种实施方式示意图；
- 图 17 (c) 为本申请实施例中对数据帧进行 CRC 的另一种实施方式示意图；
- 图 18 为本申请实施例中第三数据处理的一个实施例示意图；
- 图 19 为本申请实施例中交织缓存的一种分布示意图；
- 35 图 20 为本申请实施例中交织缓存的另一种分布示意图；
- 图 21 为本申请实施例中 PCS 和 FEC 编码的另一个实施例示意图；
- 图 22 为本申请实施例中第六比特集合的一个示意图；
- 图 23 (a) 为本申请实施例中第六比特集合的第一种分布示意图；
- 图 23 (b) 为本申请实施例中第六比特集合的第二种分布示意图；
- 40 图 23 (c) 为本申请实施例中第六比特集合的第三种分布示意图；
- 图 23 (d) 为本申请实施例中第六比特集合的第四种分布示意图；
- 图 24 (a) 为本申请实施例中第三比特子集的一个示意图；
- 图 24 (b) 为本申请实施例中第三比特子集的另一个示意图；
- 图 25 为本申请实施例中数据处理装置的第一种结构示意图；
- 45 图 26 为本申请实施例中数据处理装置的第二种结构示意图；
- 图 27 为本申请实施例中数据处理装置的第三种结构示意图；

图 28 为本申请实施例中数据处理装置的第四种结构示意图；
图 29 (a) 为本申请实施例中第三比特子集的一个示意图；
图 29 (b) 为本申请实施例中第三比特子集的一个示意图；
图 29 (c) 为本申请实施例中第三比特子集的一个示意图；
5 图 29 (d) 为本申请实施例中第三比特子集的一个示意图；
图 29 (e) 为本申请实施例中第三比特子集的一个示意图；
图 29 (f) 为本申请实施例中第三比特子集的一个示意图；
图 30 (a) 为本申请实施例中第一比特子集的一个示意图；
图 30 (b) 为本申请实施例中第一比特子集的一个示意图；
10 图 30 (c) 为本申请实施例中第一比特子集的一个示意图；
图 30 (d) 为本申请实施例中第一比特子集的一个示意图；
图 30 (e) 为本申请实施例中第一比特子集的一个示意图；
图 30 (f) 为本申请实施例中第一比特子集的一个示意图。

15 具体实施方式

本申请实施例提供了一种数据处理方法和数据处理装置，将采用 FEC 编码并结合 PCS 技术以满足更长传输距离，并使得整体数据处理操作更简单，复杂度更低，功耗更低，提升了整体性能。

需要说明的是，本申请说明书和权利要求书及上述附图中的术语“第一”、“第二”等用于区别类似的对象，而非限定特定的顺序或先后次序。应该理解，上述术语在适当情况下可以互换，以便在本申请描述的实施例能够以除了在本申请描述的内容以外的顺序实施。此外，术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含。例如，包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

图 1 为本申请实施例应用的一种通信系统示意图。如图 1 所示，在发送端，信源提供待发送的数据流，发端数据处理器接收该数据流。发端数据处理器先对其进行包含编码、交织、调制、DSP 成帧的数据处理得到符号数据流，再送入发端信号处理器进行信号处理，进而经过信道传输到达接收设备。接收设备接收到因为信道中的噪声或者其他损伤产生的失真信号后，送到收端信号处理器进行色散补偿、同步、相位恢复等操作，然后送入收端数据处理进行包含解调、解交织、译码处理后恢复出原有数据，进而将恢复的数据发给信宿。

需要说明的是，本申请实施例中涉及的公式中出现的一些运算符号，本申请实施例中出现的 $\lfloor a \rfloor$ 运算，表示对 a 进行向下取整运算，例如 $\lfloor 1.5 \rfloor = 1$ ， $\lfloor -1.5 \rfloor = -2$ 。本申请实施例中出现的 $b\%c$ 运算，表示 b 对 c 取模，当 b 为非负数时， $b\%c$ 的值取 b 除以 c 的余数，例如， $8\%3=2$ ， $2\%3=3$ 。当 b 为负数时， $b\%c$ 的值为将 b 加上 a 的整数倍成为正整数后再进行取模运算，例如， $-9\%16=7$ ， $-10\%3=2$ 。还需要说明的是，下文统一以 OFEC 编码作为 FEC 编码的一种具体实现方式进行介绍。

需要说明的是，本申请说明书和权利要求书中的比特集合、比特子集，只是为了便于描述而引入的概念。在实际应用中，数据流为一个整体并不存在划分，每个比特集合、比特子集可以视作数据流中的一个或多个比特。应理解，所述比特集合、比特子集也可以以矩阵、数组等形式呈现，具体此处不做限定。

图 2 为一种数据帧的结构示意图。如图 2 所示，数据帧 (frame) 包括多行比特，且每行包括 q 个比特，例如 $q=10280$ 。应理解，本申请不限定数据帧的具体类型。作为一个示例，一个 800ZR 帧或 FlexO-8 帧或 FlexO-8e 帧包含 512 行，每行包含 10280 比特。作为另一个示例，一个 FlexO-6 帧或 FlexO-6e 帧包含 384 行，每行包含 10280 比特。在一些具体应用中，所述整数 q 是 257 的整数倍；典型的， q 为 10280，8224，4112，或 2056 等。

需要说明的是，本申请提供的数据处理方法可以分为三部分，分别记为“第一数据处理”、“第二数据处理”和“第三数据处理”。在一种可能的实施方式中，可以先进行第一数据处理，再进行第二数据处理。在另一种可能的实施方式中，还可以先进行第一数据处理，再进行第三数据处理。下面分别对

“第一数据处理”、“第二数据处理”和“第三数据处理”进行详细介绍。

图 3 为本申请实施例第一数据处理的一个实施例示意图。如图 3 所示, 第一数据处理包含循环冗余校验 (Cyclic Redundancy Check, CRC)、插入填充 (Pad Insertion), 和加扰 (scrambling)。在实际应用中, 可以执行 CRC 和插入填充中的至少一项操作。具体地, 从数据帧中获取第一数据, 第一数据包括 r 行 q 列个比特, 其中, r 为大于 0 的整数, q 为大于 0 的整数。为了便于与下文中的 r_0 和 r_1 等进行区分, 之后统一用 r_{Frame} 来表示第一数据包括的 r 行。

以 $q=10280$ 为例, 总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}}$ 个比特。之后, 对第一数据进行 CRC 和/或插入填充比特得到第二数据。其中, 在 CRC 操作中会加入 d_{CRC} 个 CRC 校验比特, 在插入填充操作中会插入 d_{PAD} 个填充 (Pad) 比特。进而, 对第二数据进行加扰得到第三数据, 第三数据的比特数量 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}}$ 个加扰比特。其中, d_{CRC} 为大于或等于 0 的整数, d_{PAD} 为大于或等于 0 的整数, 可以记 $d_{\text{CP}} = d_{\text{CRC}} + d_{\text{PAD}}$ 。此时, CRC 校验和插入填充比特对应的冗余为 $\text{OH}_{\text{CP}} = \frac{d_{\text{scr}}}{d_{\text{in}}} - 1 = (d_{\text{CRC}} + d_{\text{PAD}})/(10280 \times r_{\text{Frame}})$ 。

作为一个示例, 当 $d_{\text{PAD}} = 0$ 时, 经过 CRC 校验的 $10280 \times r_{\text{Frame}} + d_{\text{CRC}}$ 个比特不再进行插入填充比特, 而是直接进行加扰。作为另一个示例, 为了实现更低时延更低复杂度实现, 会旁路 (bypass) CRC 校验用填充比特来代替, 即此时 $d_{\text{CRC}} = 0$ 。

需要说明的是, 在一些具体应用场景, CRC 校验采用 CRC-32。此时, $d_{\text{CRC}} = 32 \times p$, 其中 p 为大于 1 的整数且整数 r_{Frame} 能被 p 整除。更具体的, 针对从数据帧中获取的 r_{Frame} 行数据, 每 r_{Frame}/p 行总共 $10280 \times r_{\text{Frame}}/p$ 个比特进行 CRC-32 添加长度 32 比特的校验比特, 重复进行 p 次 CRC-32 添加总共 $d_{\text{CRC}} = 32 \times p$ 个比特的 CRC 校验比特。

需要说明的是, 在另一些具体应用场景, CRC 校验采用 CRC-32。此时, $d_{\text{CRC}} = 32 \times p$, 其中 p 为大于 1 的整数且整数 r_{Frame} 不能被 p 整除。考虑 $r_{F0} \times (p-1) + r_{F1} = r_{\text{Frame}}$, 其中整数 r_{F0} 大于整数 r_{F1} 。在一些具体实现中, 整数 $r_{F0} = \lfloor r_{\text{Frame}}/p \rfloor + 1$, 整数 $r_{F1} = r_{\text{Frame}} - r_{F0} \times (p-1)$, 且 $r_{F0} > r_{F1}$ 。更具体的, 针对从数据帧中获取的 r_{Frame} 行数据, 对其中最前面 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 添加长度 32 比特的校验比特得到共 $32 \times (p-1)$ 个比特的 CRC 校验比特, 对最后的 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 添加长度 32 比特的校验比特, 总共得到 $d_{\text{CRC}} = 32 \times p$ 个比特的 CRC 校验比特。需要说明的是, 在一些具体实现中, $q \times r_{F0} = 41120$, 其对 r_{Frame} 行数据中最前面 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32。所述 CRC-32 编码、检测操作可直接采用现有 800GZR 中的 CRC-32 操作, 其中现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。

图 4 为本申请实施例第二数据处理的一个实施例示意图。如图 4 所示, 从第一数据处理后输出的第三数据中获取多个比特, 并轮询送入 L 个 PCS 和 FEC 编码, L 为大于 0 的偶数, 以图 4 为例, $L=4$ 。为了便于说明, 下文都以每个 PCS 和 FEC 编码输入 k 个比特为例进行介绍, k 为大于 1 的整数。也就是说, PCS 和 FEC 编码每次都以 k 个比特为一组进行处理, 实际应用中, 每个 PCS 和 FEC 编码也可以输入 $2 \times k$ 个比特或 $4 \times k$ 个比特等, 具体此处不做限定。具体地, 对每 k 个比特先分别进行 PCS 和 FEC 编码, 再分别进行第一交织处理后得到 L 条第一比特流, 其中, FEC 编码具体可以是 OFEC 编码。之后, 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 其中, 第二交织处理具体可以是 OFEC 交织, 下文将统一以 OFEC 交织为例对第二交织处理进行介绍。接下来, 对 $L/2$ 条第二比特流进行符号映射 (symbol map) 和偏振划分 (Polarization distribution) 得到一条双偏振符号流, 例如, 符号映射和偏振划分将 t 个比特映射为 1 个双偏振符号, t 为大于 0 的整数, 例如, $t=8$ 时, 双偏振符号为双偏振 DP-16QAM 符号。进而, 对双偏振符号流进行数字信号处理 (Digital Signal Processing, DSP) 成帧, 例如, DSP 成帧将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 超帧包含 175104 个双偏振符号。

需要说明的是, 第二数据处理也可包含 6 个 PCS 和 OFEC 编码, 此时对应的第二数据处理包含 6 个第一交织处理, 3 个第二交织处理, 其中每个 PCS 和 OFEC 编码的输出送入 1 个第一交织处理, 每 2 个第一交织处理的输出送入 1 个第二交织处理, 3 个第二交织处理的输出送入符号映射进行符号调制映射。还需要说明的是, 第二数据处理也可包含 8 个 PCS 和 OFEC 编码, 此时对应的第二数据处理包含 8 个第

一交织处理, 4 个第二交织处理, 其中每个 PCS 和 OFEC 编码的输出送入 1 个第一交织处理, 每 2 个第一交织处理的输出送入 1 个第二交织处理, 4 个第二交织处理的输出送入符号映射进行符号调制映射。

下面对第二数据处理的各操作进行详细介绍。

(1) PCS 和 FEC 编码:

图 5 为本申请实施例中 PCS 和 FEC 编码的一个实施例示意图。如图 5 所示, 以其中一路输入的 k 个比特为例进行介绍。对 k 个比特中第一比特集合进行 PCS 处理得到第二比特集合。对第二比特集合和 k 个比特中除第一比特集合外的第三比特集合进行 OFEC 编码得到第四比特集合。其中, 第四比特集合包括 m_0 个第一比特子集, 每个第一比特子集包括 F_0 个比特, m_0 为大于 1 的整数, F_0 为大于 1 的偶数。例如, m_0 可以为 12 或 16, F_0 可以为 256。需要说明的是, 第二比特集合中的比特数量为 $m_0 \times F_0/2$, 第四比特集合中的比特数量为 $m_0 \times F_0$ 。需要说明的是, 上述双偏振符号中的幅度位比特来自所述第二比特集合。为了便于介绍, 下文都以 $F_0 = 256$ 为例进行介绍。

在一种可能的实施方式中, $m_0 = 16$, 第一比特集合中的比特数量为 $k-1504$, 第二比特集合中的比特数量为 2048, 第三比特集合中的比特数量为 1504, 第四比特集合中的比特数量为 4096。也就是说, 将 $k-1504$ 个比特送入 PCS 处理得到 2048 个比特, 结合 k 个比特中除去送入 PCS 处理的剩余 1504 个比特, 共有 $2048+1504=3552$ 个比特送入 OFEC 编码进行编码添加 544 个 OFEC 校验比特 (parity bits), 得到总共 4096 个 OFEC 编码后的比特。此时, 所述 OFEC 编码的编码冗余为 $4096/3552-1=15.3\%$ 。

在另一种可能的实施方式中, $m_0 = 12$, 第一比特集合中的比特数量为 $k-992$, 第二比特集合中的比特数量为 1536, 第三比特集合中的比特数量为 992, 第四比特集合中的比特数量为 3072。也就是说, 将 $k-992$ 个比特送入 PCS 处理得到 1536 个比特, 结合 k 个比特中除去送入 PCS 处理的剩余 992 个比特, 共有 $1536+992=2528$ 个比特送入 FEC 编码进行编码添加 544 个 FEC 校验比特 (parity bits), 得到总共 3072 个 FEC 编码后的比特。此时, 所述 FEC 编码的编码冗余为 $3072/2528-1=21.5\%$ 。一种具体的 FEC 码可以是所述 15.3%OH 的 OFEC 码进行缩短 (shorten) 获得所述 21.5%OH 的 FEC 码。在一些场景下, 所述缩短 OFEC 码也简称为 OFEC 码。

需要说明的是, 第一数据处理输出的第三数据的比特数量 d_{scr} 是 $L \times k$ 的整数倍。此时第一数据处理输出 d_{scr} 个比特后, 再进行一次或多次第二数据处理操作可完成对 d_{scr} 个比特的处理, 不需要做更多如缓存等其他处理, 使得整个数据处理的具体硬件实现较简单, 功耗较低。

需要说明的是, 当 $m_0 = 16$ 时, PCS 处理对应的输入比特数量 $k_{PCS} = k - 1504$ 是 2、4、8、或 16 的整数倍。考虑 PCS 处理对应的输出比特数量是 2048。通常 PCS 处理会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。PCS 处理的输出比特数量 n_{LUT} 不能太小, 否则会导致整体 PCS 处理导致性能损失。PCS 处理的输出比特数量 n_{LUT} 也不能太大, 否则不利于硬件实现。在一些具体硬件实现中, n_{LUT} 为 128、256、512 或 1024。对于 $n_{LUT} = 128$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 1504)/16$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 128$ 个比特, 连续操作 16 次上述 LUT 映射后将总共 $k - 1504$ 个比特进行 LUT 映射输出对应的 2048 个比特。在一些具体应用中, 所述查找表 LUT 处理可采用包含多个子查找表来实现, 子查找表的输入比特数可以不相同, 并且输出比特数也可以不相同。本申请中, 对所述查找表具体实现不做限定。

对于 $n_{LUT} = 256$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 1504)/8$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 256$ 个比特, 连续操作 8 次上述 LUT 映射后将总共 $k - 1504$ 个比特进行 LUT 映射输出对应的 2048 个比特。对于 $n_{LUT} = 512$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 1504)/4$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 128$ 个比特, 连续操作 4 次上述 LUT 映射后将总共 $k - 1504$ 个比特进行 LUT 映射输出对应的 2048 个比特。对于 $n_{LUT} = 1024$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 1504)/2$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 1024$ 个比特, 连续操作 2 次上述 LUT 映射后将总共 $k - 1504$ 个比特进行 LUT 映射输出对应的 2048 个比特。因此, 当 $m_0 = 16$ 时, 为了便于硬件实现, 或实现较优性能, PCS 处理对应的输入比特数量 $k - 1504$ 是 2、4、8、或 16 的整数倍。

需要说明的是, 当 $m_0 = 12$ 时, PCS 处理对应的输入比特数量 $k_{PCS} = k - 992$ 是 2、3、4、6、8、12 或 16 的整数倍。考虑 PCS 处理对应的输出比特数量是 1536。通常 PCS 处理会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。PCS 处理的输出比特数量 n_{LUT} 不能太小, 否

则会导致整体 PCS 处理导致性能损失。PCS 处理的输出比特数量 n_{LUT} 也不能太大, 否则不利于硬件实现。在一些具体硬件实现中, n_{LUT} 为 96、128、192、256、384、512 或 768。对于 $n_{LUT} = 96$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/16$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 96$ 个比特, 连续操作 16 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 128$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/12$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 128$ 个比特, 连续操作 12 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 192$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/8$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 192$ 个比特, 连续操作 8 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 256$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/6$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 256$ 个比特, 连续操作 6 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 384$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/4$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 384$ 个比特, 连续操作 4 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 512$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/3$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 512$ 个比特, 连续操作 3 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。对于 $n_{LUT} = 768$ 时, PCS 处理每次将输入的 $k_{LUT} = (k - 992)/2$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 768$ 个比特, 连续操作 2 次上述 LUT 映射后将总共 $k - 992$ 个比特进行 LUT 映射输出对应的1536个比特。因此, 当 $m_0 = 12$ 时, 为了便于硬件实现, 或实现较优性能, PCS 处理对应的输入比特数量 $k - 992$ 是 2、3、4、6、8、12 或 16 的整数倍。

图 6 为本申请实施例第四比特集合的一个示意图。如图 6 所示, 在一些应用场景中, 上述第四比特集合中的 m_0 个第一比特子集具体可以呈现为比特方阵 (square block) 的形式, 比特方阵也可以简称为方阵 (block), 每个第一比特子集可以称之为比特方阵 B。例如, 如图 6 所示, $m_0 = 16$ 个比特方阵 B 分布为 2 行 8 列。第 i_0 行第 j_0 列的方阵包含 4096 个 OFEC 编码后比特中的 256 个比特, 其中 $0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$ 。又例如, $m_0 = 12$ 个比特方阵 B 分布为 2 行 6 列。第 i_0 行第 j_0 列的方阵包含 3072 个 FEC 编码后比特中的 256 个比特, 其中 $0 \leq i_0 < 2$ 且 $0 \leq j_0 < 6$ 。进一步, 每个比特方阵 B 中的 256 个比特可以分布为 F_1 行 F_1 列, 下文都以 $F_1 = 16$ 为例进行介绍。为了描述简单, B_{i_0, j_0} 表示第 i_0 行第 j_0 列的比特方阵 B。

应理解, 上述 2 行 m_1 列个比特方阵 B 在一些具体场景下也称为所述 FEC 编码的输出分组 (output blocks), 其中 $m_0 = 2 \times m_1$ 。还应理解, 上述 2 行 m_1 列个比特方阵 B 也可认为是一个 32 行 $m_0 \times 8$ 列的比特矩阵 (matrix), 其包含 32 行, 每行包含 $m_0 \times 8$ 个比特, 则比特矩阵的第 r_0 行第 c_0 列的比特对应于比特方阵 B_{i_0, j_0} 的第 i_1 行第 j_1 列比特, 其中 $i_0 = \lfloor r_0/16 \rfloor, j_0 = \lfloor c_0/16 \rfloor, i_1 = r_0 \% 16, j_1 = c_0 \% 16$, 且 $0 \leq r_0 < 32, 0 \leq c_0 < m_0 \times 8, 0 \leq i_0 < 2, 0 \leq j_0 < 8, 0 \leq i_1 < 16$ 且 $0 \leq j_1 < 16$ 。本申请中, 有些情况下一行表示 8 个比特方阵, 有些情况下一行表示比特方阵中的一行总共 16 个比特, 有些情况下一行表示 m_0 个比特方阵组成的集合中一行总共 $m_0 \times 8$ 个比特, 可根据上下文理解以区分具体情况。

在一些具体应用中, 如图 6 所示, 第 i_0 行第 j_0 列 ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 的方阵 B_{i_0, j_0} 包含第四比特集合中的第 $512 \times j_0 + 256 \times i_0$ 个比特到第 $512 \times j_0 + 256 \times i_0 + 255$ 个比特。例如, 第 $i_0 = 0$ 行第 $j_0 = 0$ 列的方阵 $B_{0,0}$ 包含第四比特集合中的第0个比特到第255个比特。再例如, 第 $i_0 = 1$ 行第 $j_0 = 7$ 列的方阵 $B_{1,7}$ 包含第四比特集合中的第3840个比特到第4095个比特。

图 7 为本申请实施例第一比特子集的一个示意图。如图 7 所示, 每个比特方阵包含 16 行 16 列总共 256 个比特, 比特方阵中第 i_1 行第 j_1 列 ($0 \leq i_1 < 16$ 且 $0 \leq j_1 < 16$) 的比特对应 256 个比特中的第 $16 \times i_1 + j_1$ 个比特。例如, 比特方阵中第 $i_1 = 1$ 行第 $j_1 = 0$ 列的比特对应 256 个比特中的第16个比特。再例如, 比特方阵中第 $i_1 = 2$ 行第 $j_1 = 3$ 列的比特对应 256 个比特中的第35个比特。

更具体的, 第 i_0 行第 j_0 列 ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 的比特方阵 B_{i_0, j_0} 中第 i_1 行第 j_1 列 ($0 \leq i_1 < 16$ 且 $0 \leq j_1 < 16$) 的比特 $B_{i_0, j_0}(i_1, j_1)$ 对应第四比特集合中的第 $512 \times j_0 + 256 \times i_0 + 16 \times i_1 + j_1$ 个比特。

例如, 第 $i_0 = 0$ 行第 $j_0 = 0$ 列的比特方阵 $B_{0,0}$ 中第 $i_1 = 1$ 行第 $j_1 = 0$ 列的比特 $B_{0,0}(1,0)$ 对应第四比特集合中的第16个比特。再例如, 第 $i_0 = 1$ 行第 $j_0 = 7$ 列的比特方阵 $B_{1,7}$ 中第 $i_1 = 2$ 行第 $j_1 = 3$ 列的比特 $B_{1,7}(2,3)$ 对

应第四比特集合中的第3875个比特。

(2) 第一交织处理:

对 PCS 处理和 FEC 编码后的第四比特集合进行第一交织处理得到第五比特集合, 具体地, 第一交织处理对第四比特集合中的比特顺序进行重排序。其中, 第五比特集合包括 m_0 个第二比特子集, 每个第二比特子集包括 256 个比特。应理解, 如图 4 所示, 送入第二交织处理的每条第一比特流可以包括多个第五比特集合。

图 8 为本申请实施例第五比特集合的一个示意图。如图 8 所示, 以 $m_0=16$ 为例, 与上述第四比特集合中的 16 个第一比特子集类似, 第五比特集合中的 16 个第二比特子集也可以呈现为比特方阵(square block) 的形式, 每个第二比特子集可以称之为比特方阵 T, 每个比特方阵 T 中的 256 个比特分布为 16 行 16 列。例如, 16 个比特方阵 T 分布为 2 行 8 列。第 i_0 行第 j_0 列的方阵包含 4096 个第一交织处理后比特中的 256 个比特, 其中 $0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$ 。为了描述简单, T_{i_0, j_0} 表示第 i_0 行第 j_0 列的方阵。同样上述 2 行 m_1 列个比特方阵 T 也可认为是一个 32 行 $m_0 \times 8$ 列的比特矩阵(matrix), 其包含 32 行, 每行包含 $m_0 \times 8$ 个比特, 则比特矩阵的第 r_1 行第 c_1 列的比特对应于比特方阵 T_{i_0, j_0} 的第 i_1 行第 j_1 列的比特, 其中 $i_0 = [r_1/16], j_0 = [c_1/16], i_1 = r_1 \% 16, j_1 = c_1 \% 16$, 且 $0 \leq r_1 < 32, 0 \leq c_1 < m_0 \times 8, 0 \leq i_0 < 2, 0 \leq j_0 < 8, 0 \leq i_1 < 16$ 且 $0 \leq j_1 < 16$ 。应理解, 第一交织处理输出的 2 行 8 列个比特方阵 T 在一些具体场景下也称为第一交织处理的输出分组(output blocks)。

需要说明的是, 每个第二比特子集中的其中 128 个比特来自第二比特集合, 每个第二比特子集中的另外 128 个比特来自第三比特集合和/或 OFEC 编码的校验比特。作为一个示例, 经 PCS 处理的 2048 个比特, 再经过 OFEC 编码, 并进行第一交织处理后, 分布在 16 个比特方阵 T 中, 且分布到每个包含 256 个比特方阵 T 中的 128 个比特。

在一些具体应用中, 第二比特子集的每行 16 个比特中的其中 8 个比特来自第二比特集合, 第二比特子集的每行 16 个比特中的另外 8 个比特来自第三比特集合和/或 OFEC 编码的校验比特。

图 9 (a) 为本申请实施例第二比特子集的一个示意图。如图 9 (a) 所示, 在一种可能的实施方式中, 第二比特子集中第 i_2 行第 $(\lfloor \frac{j_2}{2} \rfloor \times 4 + j_2 \% 2 + i_2 + 2) \% 16$ 列的比特来自第二比特集合, $0 \leq i_2 < 16, 0 \leq j_2 < 8$, 即图 9 (a) 中阴影部分的比特来自第二比特集合的 2048 个比特中的 1 个比特。图 9 (b) 为本申请实施例第二比特子集的另一个示意图。如图 9 (b) 所示, 在另一种可能的实施方式中, 第二比特子集中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自第二比特集合, $0 \leq i_2 < 16, 0 \leq j_2 < 8$, 即图 9 (b) 中阴影部分的比特来自第二比特集合的 2048 个比特中的 1 个比特。

需要说明的是, 第四比特集合中 m_1 个第一比特子集中共 $m_1 \times 256$ 个比特来自第二比特集合, $m_0=2 \times m_1$ 。请参阅图 6, 以 $m_0=16, m_1=8$ 为例, 2 行 8 列共 16 个比特方阵 B 中有 8 个比特方阵 B 来自第二比特集合的 2048 个比特。更进一步, 第四比特集合的 4096 个比特中第 0 比特-2047 比特来自第二比特集合的 2048 个比特。

在一种可能的实施方式中, 第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中, $0 \leq r_0 < 32, 0 \leq r_1 < 32$ 。

图 10 为本申请实施例第一交织处理的第一种实施方式示意图。如图 10 所示, 第四比特集合(包括 16 个比特方阵 B) 中 4096 个比特分布为 32 行 128 列, 第五比特集合(包括 16 个比特方阵 T) 中 4096 个比特分布为 32 行 128 列。第五比特集合中第 r_1 行的 128 个比特来自第四比特集合中第 r_0 行的 128 个比特。更具体的, 当 $r_1 \% 4 = 0$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 2、3、6、7、10、11、14、15、...、114、115、118、119、122、123、126、127 比特总共 64 个比特来自第四比特集合中第 r_0 行 128 个比特中来自经 PCS 处理的 64 个比特。当 $r_1 \% 4 = 1$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 0、3、4、7、8、11、12、15、...、112、115、116、119、120、123、124、127 比特总共 64 个比特来自第四比特集合中第 r_0 行 128 个比特中来自经 PCS 处理的 64 个比特。当 $r_1 \% 4 = 2$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 0、1、4、5、8、9、12、13、...、112、113、116、117、120、121、124、125 比特总共 64 个

比特来自第四比特集合中第 r_0 行 128 个比特中来自经 PCS 处理的 64 个比特。当 $r_1 \% 4 = 3$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 1、2、5、6、9、10、13、14、...、113、114、117、118、121、122、125、126 比特总共 64 个比特来自第四比特集合中第 r_0 行 128 个比特中来自经 PCS 处理的 64 个比特。

作为一个示例, m_0 个第一比特子集分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第一比特子集 B_{0,m_2} 和
5 第 1 行第 m_2 列的第一比特子集 B_{1,m_2} 来自第二比特集合, $0 \leq m_2 < m_1/2$ 。

图 11(a)为本申请实施例中多个第一比特子集的一种分布示意图。如图 11(a)所示, 以 $m_0 = 16$ 为例, 8 个阴影背景的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 来自第二比特集合的 2048 个比特, 8 个无阴影背景的比特方阵 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 来自未经 PCS 处理的 1504 个比特和 544 个 OFEC 校验比特。以 16 个比特方阵 B 为整体来看, 第 r_0 行的 128 个比特中前面 64 个比特为
10 来自第二比特集合的 2048 个比特中的 64 个比特, 其中, $0 \leq r_0 < 32$ 。

以图 11(a) 为例, 第一交织处理的一种具体交织规则如下: 第五比特集合 (包括 m_0 个比特方阵 T)

中第 r_1 行的 $m_0 \times 8$ 个比特中第 $\left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% (m_0 \times 8)$ 个比特来自第四比特集合 (包括 m_0 个比特方阵 B) 中第 r_0 行的第 0 比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < (m_0 \times 4)$ 。
下面进一步提供几种更具体地示例。

15 示例 1: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)$ 。

20 示例 2: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor$ 。

25 示例 3: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times m_0 \times 4$ 。

30 示例 4: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + \left\lfloor \frac{c_0 \% (m_0 \times 4)}{16} \right\rfloor \times 16 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times m_0 \times 4$ 。

35 以图 11(a) 为例, 第一交织处理的另一种具体交织规则如下: 第五比特集合 (包括 m_0 个比特方阵 T) 中第 r_1 行的 $m_0 \times 8$ 个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合 (包括 m_0 个比特方阵 B)

中第 r_0 行的第 0 比特到第 $(m_0 \times 4 - 1)$ 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < (m_0 \times 4)$ 。下面进一步提供几种更具体地示例。

示例 5: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$5 \quad c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = c_0\%(m_0 \times 4) * 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right)$

示例 6: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$10 \quad c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍, $c = c_0\%(m_0 \times 4) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor$

示例 7: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$15 \quad c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍,

$$c = (c_0\%8) \times 2 + \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor\right) + 16 \times \left\lfloor \frac{c_0\%(m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0\%16}{8} \right\rfloor$$

示例 8: 第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特中第 c_0 个比特;

$$20 \quad c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < (m_0 \times 8)$, $0 \leq c_1 < (m_0 \times 8)$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍,

$$c = (c_0\%8) \times 2 + \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor + 16 \times \left\lfloor \frac{c_0\%(m_0 \times 4)}{16} \right\rfloor + (m_0 \times 4) \times \left\lfloor \frac{c_0\%16}{8} \right\rfloor$$

以图 11 (a) 为例, 在一种可能的实施方式中, 满足如下条件:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特。

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特。

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特。

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特。

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特。

第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特。

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特。

第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

5 以图 11 (a) 为例, 在另一种可能的实施方式中, 满足如下条件:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特。

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特。

10 第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特。

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特。

15 第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特。

第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特。

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特。

20 第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

以图 11 (a) 为例, 在又一种可能的实施方式中, 满足如下条件:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 0 行第 1 列的第二比特子集 $T_{0,1}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 4 列的第一比特子集 $B_{0,4}$ 中共 512 个比特,

25 第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 0 行第 3 列的第二比特子集 $T_{0,3}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 5 列的第一比特子集 $B_{0,5}$ 中共 512 个比特,

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 0 行第 5 列的第二比特子集 $T_{0,5}$ 中共 512 个比特来自第 0 行第 2 列的第一比特子集 $B_{0,2}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,

30 第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 0 行第 7 列的第二比特子集 $T_{0,7}$ 中共 512 个比特来自第 0 行第 3 列的第一比特子集 $B_{0,3}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,

第 1 行第 0 列的第二比特子集 $T_{1,0}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 4 列的第一比特子集 $B_{1,4}$ 中共 512 个比特,

第 1 行第 2 列的第二比特子集 $T_{1,2}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 5 列的第一比特子集 $B_{1,5}$ 中共 512 个比特,

35 第 1 行第 4 列的第二比特子集 $T_{1,4}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 2 列的第一比特子集 $B_{1,2}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,

第 1 行第 6 列的第二比特子集 $T_{1,6}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 3 列的第一比特子集 $B_{1,3}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

40 图 11(b)为本申请实施例中多个第一比特子集的另一种分布示意图。如图 11(b)所示, 以 $m_0 = 16$ 为例, 8 个阴影背景的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 来自第二比特集合的 2048 个比特, 8 个无阴影背景的比特方阵 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 来自未经 PCS 处理的 1504 个比特和 544 个 OFEC 校验比特。以 16 个比特方阵 B 为整体来看, 第 r_0 行 128 个比特中第 0 至第 31 个比特和第 64 至第 95 个比特为来自第二比特集合的 2048 个比特中的 64 个比特, 其中, $0 \leq r_0 < 32$ 。

45 以图 11 (b) 为例, 第一交织处理的一种具体交织规则如下: 第五比特集合 (包括 m_0 个比特方阵 T)

中第 r_1 行的 128 个比特中第 $\left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 128$ 个比特来自第四比特集合（包括 m_0 个比特方阵B）中第 r_0 行的第0比特到第31比特或第64比特到第95比特中的一个比特，其中， $0 \leq \tilde{c}_0 < 64$ 。下面进一步提供几种更具体地示例。

5 示例 9：第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特；

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中， $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是4的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是4的整数倍， $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

10 示例 10：第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特；

$$c_1 = (c \% \theta + (r_1 + 2 + 4 * \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中， $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是4的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是4的整数倍， $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

15 示例 11：第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特；

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中， $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是4的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是4的整数倍， $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

20 示例 12：第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特；

$$c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中， $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是4的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是4的整数倍， $c = \left\lfloor \frac{c_0 \% 8}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2\right) \times 16$ 。

25 以图 11 (b) 为例，第一交织处理的一种具体交织规则如下：第五比特集合（包括 m_0 个比特方阵T）中第 r_1 行的 128 个比特中第 $(\tilde{c}_0 \times 2 - i_2 \% 2 + 1)$ 个比特来自第四比特集合（包括 m_0 个比特方阵B）中第 r_0 行的第0比特到第31比特或第64比特到第95比特中的一个比特，其中， $0 \leq \tilde{c}_0 < 64$ 。下面进一步提供几种更具体地示例。

30 示例 13：第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特；

$$c_1 = (c \% \theta + (r_1 + 2 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中， $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是4的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是2的整数倍， $c = (c_0 \% 32) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right)\right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$

示例 14: 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整

5 数倍, $c = (c_0 \% 32) \times 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$

示例 15: 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

10 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍,

$$c = (c_0 \% 8) \times 2 + \left(1 - \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \right) + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) \times 16$$

示例 16: 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特;

$$c_1 = (c\% \theta + (r_1 + 1 + 2 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta;$$

15 其中, $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 2 的整数倍,

$$c = (c_0 \% 8) * 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) + \left\lfloor \frac{c_0}{64} \right\rfloor * 64 + \left\lfloor \frac{c_0 \% 16}{8} \right\rfloor * 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor \% 2 \right) * 16$$

以图 11 (b) 为例, 在一种可能的实施方式中, 满足如下条件:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特。

20 第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 2 列的第一比特子集 $B_{1,2}$ 中共 512 个比特。

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 3 列的第一比特子集 $B_{0,3}$ 中共 512 个比特。

25 第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 3 列的第一比特子集 $B_{1,3}$ 中共 512 个比特。

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 0 行第 4 列的第一比特子集 $B_{0,4}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特。

第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 4 列的第一比特子集 $B_{1,4}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特。

30 第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 0 行第 5 列的第一比特子集 $B_{0,5}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特。

第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

以图 11 (b) 为例, 在另一种可能的实施方式中, 满足如下条件:

35 第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 1 行第 0 列的第二比特子集 $T_{1,0}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特。

第 0 行第 1 列的第二比特子集 $T_{0,1}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 0 行第

1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 3 列的第一比特子集 $B_{0,3}$ 中共 512 个比特。

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 1 行第 2 列的第二比特子集 $T_{1,2}$ 中共 512 个比特来自第 0 行第 4 列的第一比特子集 $B_{0,4}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特。

第 0 行第 3 列的第二比特子集 $T_{0,3}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 0 行第 5 列的第一比特子集 $B_{0,5}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特。

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 1 行第 4 列的第二比特子集 $T_{1,4}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 2 列的第一比特子集 $B_{1,2}$ 中共 512 个比特。

第 0 行第 5 列的第二比特子集 $T_{0,5}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 3 列的第一比特子集 $B_{1,3}$ 中共 512 个比特。

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 1 行第 6 列的第二比特子集 $T_{1,6}$ 中共 512 个比特来自第 1 行第 4 列的第一比特子集 $B_{1,4}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特。

第 0 行第 7 列的第二比特子集 $T_{0,7}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

以图 11 (b) 为例, 在又一种可能的实施方式中, 满足如下条件:

第 0 行第 0 列的第二比特子集 $T_{0,0}$ 和第 0 行第 1 列的第二比特子集 $T_{0,1}$ 中共 512 个比特来自第 0 行第 0 列的第一比特子集 $B_{0,0}$ 和第 0 行第 2 列的第一比特子集 $B_{0,2}$ 中共 512 个比特,

第 0 行第 2 列的第二比特子集 $T_{0,2}$ 和第 0 行第 3 列的第二比特子集 $T_{0,3}$ 中共 512 个比特来自第 0 行第 1 列的第一比特子集 $B_{0,1}$ 和第 0 行第 3 列的第一比特子集 $B_{0,3}$ 中共 512 个比特,

第 0 行第 4 列的第二比特子集 $T_{0,4}$ 和第 0 行第 5 列的第二比特子集 $T_{0,5}$ 中共 512 个比特来自第 0 行第 4 列的第一比特子集 $B_{0,4}$ 和第 0 行第 6 列的第一比特子集 $B_{0,6}$ 中共 512 个比特,

第 0 行第 6 列的第二比特子集 $T_{0,6}$ 和第 0 行第 7 列的第二比特子集 $T_{0,7}$ 中共 512 个比特来自第 0 行第 5 列的第一比特子集 $B_{0,5}$ 和第 0 行第 7 列的第一比特子集 $B_{0,7}$ 中共 512 个比特,

第 1 行第 0 列的第二比特子集 $T_{1,0}$ 和第 1 行第 1 列的第二比特子集 $T_{1,1}$ 中共 512 个比特来自第 1 行第 0 列的第一比特子集 $B_{1,0}$ 和第 1 行第 2 列的第一比特子集 $B_{1,2}$ 中共 512 个比特,

第 1 行第 2 列的第二比特子集 $T_{1,2}$ 和第 1 行第 3 列的第二比特子集 $T_{1,3}$ 中共 512 个比特来自第 1 行第 1 列的第一比特子集 $B_{1,1}$ 和第 1 行第 3 列的第一比特子集 $B_{1,3}$ 中共 512 个比特,

第 1 行第 4 列的第二比特子集 $T_{1,4}$ 和第 1 行第 5 列的第二比特子集 $T_{1,5}$ 中共 512 个比特来自第 1 行第 4 列的第一比特子集 $B_{1,4}$ 和第 1 行第 6 列的第一比特子集 $B_{1,6}$ 中共 512 个比特,

第 1 行第 6 列的第二比特子集 $T_{1,6}$ 和第 1 行第 7 列的第二比特子集 $T_{1,7}$ 中共 512 个比特来自第 1 行第 5 列的第一比特子集 $B_{1,5}$ 和第 1 行第 7 列的第一比特子集 $B_{1,7}$ 中共 512 个比特。

需要说明的是, 如图 11(b) 所示, 8 个阴影背景的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 来自第二比特集合的 2048 个比特。结合图 5, 对于如图 11 (b) 所示, 所述来自第二比特集合的 2048 个比特在所述 FEC 编码的码字中位置是不连续的。此时, 所述第二比特集合和所述第三比特集合在送入所述 FEC 编码前还经过了编码前交织处理以打乱顺序, 使其编码前的 3552 个比特的排放顺序如下: 先是来自第二比特集合的连续 1024 个比特, 接着是来自第三比特集合的连续 1024 个比特, 再接着是来自第二比特集合的连续 1024 个比特, 最后是来自第三比特集合的连续 480 个比特。

需要说明的是, 所述编码前交织处理是可选的, 但是所述第一交织处理是不可或缺的。比如对应如图 11 (a) 所示, 所述第二比特集合和所述第三比特集合不经过编码前交织处理, 但编码后的第四比特集合仍需送入第一交织处理; 比如对应如图 11 (b) 所示, 所述第二比特集合和所述第三比特集合经过编码前交织处理, 编码后的第四比特集合需送入第一交织处理进行打乱顺序。

需要说明的是, 在一些具体应用中, 所述第二比特集合和所述第三比特集合经过编码前交织处理, 编码后的第四比特集合也送入第一交织处理进行打乱顺序。编码后的第四比特集合中的某几个第一比特子集已满足如图 9 (a) 或图 9 (b) 所示图样时, 第一交织处理可仅对不满足如图 9 (a) 或图 9 (b) 所示图样的剩余几个第一比特子集进行打乱顺序, 使得编码后的第四比特集合经过第一交织处理后的第五比特集合中的所有第二比特子集满足如图 9 (a) 或图 9 (b) 所示图样。也就是说, 所述第一交织处理可对所述第四比特集合中的部分比特进行交织处理, 此处不做具体限定。可选地, 在一些场景下, 如果

编码后的第四比特集合中的某几个第一比特子集已满足如图 9 (a) 或图 9 (b) 所示图样, 仍可以对所有的第一比特子集进行第一交织处理, 只要使得第一交织处理后的第五比特集合中的所有第二比特子集满足如图 9 (a) 或图 9 (b) 所示图样即可。

下面给出几种编码前交织和对应第一交织的实施方案。

5 图 21 为本申请实施例中 PCS 和 FEC 编码的另一个实施例示意图。如图 21 所示, 对所述第二比特集合和所述第三比特集合先进行编码前交织得到所述的第六比特集合, 再对第六比特集合进行 FEC 编码得到第四比特集合。第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数。其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数。

10 应理解, 上述“第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列”只是为了便于描述而引入的概念, 在实际应用中, 第六比特集合为一个整体并不存在“2 行 m_4 列”的分布, 每个第三比特子集可以视作第六比特集合中的多个比特。

15 图 22 为本申请实施例中第六比特集合的一个示意图。作为一个示例, 所述的第六比特集合包含 $m_3 = 14$ 个第三比特子集。在一些应用场景中, 上述第六比特集合中的 m_3 个第三比特子集具体可以呈现为比特方阵 (square block) 的形式, 比特方阵也可以简称为方阵 (block), 每个第三比特子集可以称之为比特方阵 I。例如图 22 所示, $m_3 = 14$ 个比特方阵 I 分布为 2 行 7 列, 为了描述简单, I_{i_0, j_0} 表示第六比特集合的第 i_0 行第 j_0 列的比特方阵 I。当 $0 \leq i_0 < 2$ 且 $0 \leq j_0 < 6$ 时, I_{i_0, j_0} 包含 $F_0 = 256$ 比特, 排列为 16 行 16 列, 而 $I_{0,6}$ 与 $I_{1,6}$ 分别包含 $F_2 = 240$ 比特, 排列为 16 行 15 列。

20 应理解, 上述 2 行 m_4 列个比特方阵 I 在一些具体场景下也称为所述 FEC 编码的输入分组 (input blocks), 其中 $m_3 = 2 \times m_4$ 。还应理解, 上述 2 行 m_4 列个比特方阵 I 也可认为是一个 32 行 $m_3 \times 8 - 1$ 列的比特矩阵 (matrix), 其包含 32 行, 每行包含 $m_3 \times 8 - 1$ 个比特, 则比特矩阵的第 r_0 行第 c_0 列的比特对应于比特方阵 I_{i_0, j_0} 的第 i_1 行第 j_1 列比特, 其中 $i_0 = \lfloor r_0 / 16 \rfloor$, $j_0 = \lfloor c_0 / 16 \rfloor$, $i_1 = r_0 \% 16$, $j_1 = c_0 \% 16$,

25 $0 \leq r_0 < 32$, $0 \leq c_0 < m_3 \times 8 - 1$, $0 \leq i_0 < 2$, $0 \leq j_0 < 7$, $0 \leq i_1 < 16$, $0 \leq j_1 < 16$ 。本申请中, 有些情况下一行表示 8 个比特方阵, 有些情况下一行表示比特方阵中的一行总共 16 个比特, 有些情况下一行表示 m_3 个比特方阵组成的集合中的一行总共 $m_3 \times 8 - 1$ 个比特, 可根据上下文理解以区分具体情况。

30 在一些具体应用中, 如图 22 所示, 第 i_0 行第 j_0 列 ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 6$) 的方阵 I_{i_0, j_0} 包含第六比特集合中的第 $512 \times j_0 + 256 \times i_0$ 个比特到第 $512 \times j_0 + 256 \times i_0 + 255$ 个比特。例如, 方阵 $I_{0,0}$ 包含第六比特集合中的第 0 个比特到第 255 个比特。再例如, 方阵 $I_{1,0}$ 包含第六比特集合中的第 256 个比特到第 511 个比特。由于方阵 $I_{0,6}$ 和方阵 $I_{1,6}$ 都是 16 行 15 列, 则其分别包含第六比特集合中的第 3072 个比特到第 3311 个比特以及第六比特集合中的第 3312 个比特到第 3551 个比特。

35 图 23 (a) 为本申请实施例中第六比特集合的第一种分布示意图。如图 23 (a) 所示, 以 $m_2 = 14$ 为例, 第六比特集合中比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,4}$ 、 $I_{0,5}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,4}$ 、 $I_{1,5}$ 共 2048 比特来自第二比特集合。比特方阵 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,6}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{0,6}$ 所有比特来自第三比特集合。然后第六比特集合经过 FEC 编码后得到第四比特集合。则对应的第四比特集合分布如图 11 (b) 所示, 在此情景下第一交织可以按照示例 9-示例 12 中的任一方案得到所述的第五比特集合, 且第五比特集合中的任一第二比特子集如图 9 (a) 所示。或者, 第一交织可以按照示例 13-示例 16 中的任一方案得到所述的第五比特集合, 且第五比特集合中的任一第二比特子集如图 9 (b) 所示。

40 图 23 (b) 为本申请实施例中第六比特集合的第二种分布示意图。如图 23 (b) 所示, 以 $m_2 = 14$ 为例, 第六比特集合中比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 中的每个比特方阵中的一半比特 (128 比特) 来自第二比特集合, 另外一半比特 (128 比特) 来自未经过 PCS 处理的第三比特集合。比特方阵 $I_{0,4}$ 、 $I_{0,5}$ 、 $I_{1,4}$ 、 $I_{1,5}$ 所有比特来自第二比特集合。比特方阵 $I_{0,6}$ 、 $I_{1,6}$ 来自未经 PCS 处理的第三比特集合。

图 24 (a) 为本申请实施例中第三比特子集的一个示意图。图 24 (b) 为本申请实施例中第三比特子集的另一个示意图。基于图 23(b) 所示的第六比特集合, 进一步, 比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 满足图 24 (a) 所示的图样, 即比特方阵 I_{i_0,j_0} 第 i_1 行第 $\left(\left\lfloor \frac{i_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列比特来自所示第

二比特集合, 而比特方阵 I_{i_0,j_0} 的其他比特来自未经过 PCS 处理的第三比特集合, $0 \leq i_0 < 2$, $0 \leq j_0 <$

- 5 4, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$; 或者, 比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 满足图 24 (b) 所示的图样, 即比特方阵 I_{i_0,j_0} 第 i_1 行第 $(j_1 \times 2)$ 列比特来自所示第二比特集合, 而比特方阵 I_{i_0,j_0} 的其他比特来自未经过 PCS 处理的第三比特集合, $0 \leq i_0 < 2$, $0 \leq j_0 < 4$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$ 。

- 图 11 (c) 为本申请实施例中多个第一比特子集的另一种分布示意图。基于图 23 (b) 所示的第六比特集合经过 FEC 编码得到第四比特集合, 则对应的第四比特集合分布如图 11 (c) 所示。具体为比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 中的每个比特方阵中的一半比特 (128 比特) 来自第二比特集合, 另外一半比特 (128 比特) 来自未经过 PCS 处理的第三比特集合。比特方阵 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 所有比特来自第二比特集合。比特方阵 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 来自未经 PCS 处理的第三比特集合和/或 OFEC 校验比特。进一步, 第四比特集合中的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 的比特排列方式满足图 9(a) 或者图 9(b) 所示。具体地, 当第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 如图 24 (a) 所示, 则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 如图 9 (a) 所示。当第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 如图 24 (b) 所示, 则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 如图 9 (b) 所示。在此情景下第一交织只需要对第四比特集合中的比特方阵 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 共 8 个比特方阵进行第一交织得到第五比特集合中比特方阵 $T_{0,4}$ 、 $T_{0,5}$ 、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,4}$ 、 $T_{1,5}$ 、 $T_{1,6}$ 、 $T_{1,7}$, 而 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 按照原有的顺序输出得到第五比特集合中比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 。

- 以图 11 (c) 为例, 且第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 如图 9 (a) 所示, 第一交织处理的一种具体交织规则如下: 第五比特集合 (包括 m_0 个比特方阵 T) 中第 r_1 行的 128 个比特中的第 0 比特到第 63 比特与第四比特集合第 r_0 行的第 0 比特到第 63 比特一一对应。且第五比特集合中第 r_1 行的 128 个比特中第 $64 + \left(\left\lfloor \frac{\tilde{c}_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4\right) \% 64$ 比特来自第四比特集合 (包括 m_0 个比特方阵 B) 中第 r_0 行的第 64 比特到第 95 比特中的一个比特, 其中, $0 \leq \tilde{c}_0 < 31$ 。下面进一步提供几种更具体地示例。

- 示例 17: 当 $0 \leq c_1 < 64$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特。当 $64 \leq c_1 < 128$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特。

$$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta; \text{ 其中, } 64 \leq c_0 < 128, 64 \leq c_1 < 128, \theta \text{ 是 } 4 \text{ 的整数}$$

倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$ 。

- 示例 18: 当 $0 \leq c_1 < 64$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特。当 $64 \leq c_1 < 128$ 第五时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特。

$$c_1 = (c \% \theta + (r_1 + 2 + 4 * \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta. \text{ 其中, } 64 \leq c_0 < 128, 64 \leq c_1 < 128, \theta \text{ 是 } 4 \text{ 的整}$$

数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + 64$ 。

示例 19: 当 $0 \leq c_1 < 64$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合

中第 r_0 行的 128 个比特中第 c_1 个比特。当 $64 \leq c_1 < 128$ 第五时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特。

$$c_1 = (c\% \theta + (r_1 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta。其中, 64 \leq c_0 < 128, 64 \leq c_1 < 128, \theta 是 4 的整数倍,$$

且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0\%8}{2} \right\rfloor \times 4 + c_0\%2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor\%2\right) \times 2 + 64 +$

$$5 \left\lfloor \frac{c_0\%16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor\%2\right) \times 16。$$

示例 20: 当 $0 \leq c_1 < 64$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $64 \leq c_1 < 128$ 第五时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特。

$$c_1 = (c\% \theta + (r_1 + 2 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta。其中, 64 \leq c_0 < 128, 64 \leq c_1 < 128, \theta 是 4 的整$$

10 数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0\%8}{2} \right\rfloor \times 4 + c_0\%2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor\%2\right) \times 2 + 64 +$

$$\left\lfloor \frac{c_0\%16}{8} \right\rfloor \times 32 + \left(\left\lfloor \frac{c_0}{16} \right\rfloor\%2\right) \times 16。$$

图 23 (c) 为本申请实施例中第六比特集合的第三种分布示意图。如图 23 (c) 所示, 第六比特集合中比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 中的每个比特方阵中的一半比特 (128 比特) 来自第二比特集合, 另外一半比特 (128 比特) 来自未经过 PCS 处理的第三比特集合。比特方阵 $I_{0,5}$ 、比特方阵 $I_{1,5}$ 、比特方阵 $I_{0,6}$ 第 0 列到第 7 列以及比特方阵 $I_{1,6}$ 的第 0 列到第 7 列的所有比特来自第二比特集合。比特方阵 $I_{0,6}$ 的第 8 列到第 14 列以及比特方阵 $I_{1,6}$ 的第 8 列到第 14 列所示比特来自未经 PCS 处理的第三比特集合。进一步, 比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 都满足图 24 (a) 所示的图样, 即比特方阵 I_{i_0,j_0} 第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1\%2 + 2 - i_1\%2\right)$ 列比特来自所示第二比特集合, 而比特方阵 I_{i_0,j_0} 的其他比特来自为经过 PCS 处理的第三比特集合, $0 \leq i_0 < 2$, $0 \leq j_0 < 5$, $0 \leq i_1 < 16$,

20 $0 \leq j_1 < 8$; 或者, 比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 都满足图 24 (b) 所示的图样, 即比特方阵 I_{i_0,j_0} 第 i_1 行第 $(j_1 \times 2)$ 列比特来自所示第二比特集合, 而比特方阵 I_{i_0,j_0} 的其他比特来自为经过 PCS 处理的第三比特集合, $0 \leq i_0 < 2$, $0 \leq j_0 < 5$, $0 \leq i_1 < 16$, $0 \leq j_1 < 8$ 。

图 11 (d) 为本申请实施例中多个第一比特子集的另一种分布示意图。基于图 23 (c) 所示的第六比特集合经过 FEC 编码得到第四比特集合, 则对应的第四比特集合分布如图 11 (d) 所示, 具体为第四比特集合中的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 中的每个比特方阵中的一半比特 (128 比特) 来自第二比特集合, 另外一半比特 (128 比特) 来自未经过 PCS 处理的第三比特集合。比特方阵 $B_{0,5}$ 、比特方阵 $B_{1,5}$ 以及比特方阵 $B_{0,6}$ 、比特方阵 $B_{1,6}$ 中的阴影部分所有比特来自第二比特集合。比特方阵 $B_{0,6}$ 、比特方阵 $B_{0,6}$ 中非阴影部分的比特以及比特方阵 $B_{0,7}$ 、比特方阵 $B_{1,7}$ 的所有比特来自未经 PCS 处理的第三比特集合和/或 FEC 校验比特。进一步, 第四比特集合中的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 中来自第二比特集合的比特和来自未经 PCS 处理的第三比特集合的比特排列方式满足图 9(a)或者图 9(b)所示。具体地, 当第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 如图 24(a)所示, 则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 如图 9(a)所示。当第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 如图 24 (b) 所示, 则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 如图 9 (b) 所示。在此情景下第一交织只需要对第四比特集合中 $B_{0,5}$ 、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,5}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 共 6 个比特方阵进行第一交织得到第五比特集合中 $T_{0,5}$ 、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,5}$ 、 $T_{1,6}$ 、 $T_{1,7}$, 而 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 按照原有的顺序输出得到第五比特集合中的比特方阵 $T_{0,0}$ 、 $T_{0,1}$ 、

$T_{0,2}$ 、 $B_{0,3}$ 、 $T_{0,4}$ 、 $T_{1,0}$ 、 $T_{1,1}$ 、 $T_{1,2}$ 、 $T_{1,3}$ 、 $T_{1,4}$ 。

以图 11(d)为例,且第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 如图 9

(a)所示,第一交织处理的一种具体交织规则如下:第五比特集合(包括 m_0 个比特方阵 T)中第 r_1 行的 128 个比特中的第 0 比特到第 79 比特与第四比特集合(包括 m_0 个比特方阵 B)第 r_0 行的第 0 比特到

5 第 79 比特一一对应。当 $0 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 8$ 时,第五比特集合(包括 m_0 个比特方阵 T)第 r_1 行的 128 个比特中

第 $80 + \left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4 \right) \% 48$ 比特来自第四比特集合(包括 m_0 个比特方阵 B)中第 r_0 行的

第 80 比特到第 103 比特中的任一个比特。当 $8 \leq \left\lfloor \frac{r_0}{16} \right\rfloor < 16$ 时,第五比特集合(包括 m_0 个比特方阵 T)

第 r_1 行的 128 个比特中第 $80 + \left(\left\lfloor \frac{c_0}{2} \right\rfloor \times 4 + \tilde{c}_0 \% 2 + (r_1 + 2) \% 4 \right) \% 48$ 比特来自第四比特集合(包括 m_0 个

比特方阵 B)中第 r_0 行的第 80 比特到第 95 比特以及第 104 比特到第 111 比特中的任一个比特。其中,

10 $0 \leq \tilde{c}_0 < 24$ 。下面进一步提供几种更具体地示例。

示例 21: 当 $0 \leq c_1 < 80$ 时,第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特。当 $80 \leq c_1 < 128$ 时,所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特。 $c_1 = 80 + (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$ 。其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍,且 48 能被 θ 整除, Δ 为整

15 数, β 是 4 的整数倍。 $c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(1 - \left\lfloor \frac{c_2}{24} \right\rfloor \right) \times 2$, $c_2 = (r_0 \% 16) \wedge c_0$ 。

示例 22: 当 $0 \leq c_1 < 80$ 时,所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特; 当 $80 \leq c_1 < 128$ 时,所述第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特。 $c_1 = 80 + (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$ 。其中, $0 \leq c_0 < 48$, $80 \leq c_1 < 128$, θ 是 4 的整数倍,且 48 能

20 被 θ 整除, Δ 为整数, β 是 4 的整数, $c = \left\lfloor \frac{c_2 \% 24}{2} \right\rfloor \times 4 + c_2 \% 2 + \left(\left\lfloor \frac{c_2}{24} \right\rfloor \right) \times 2$, $c_2 = (r_0 \% 16) \wedge c_0$ 。

图 23(d)为本申请实施例中第六比特集合的第四种分布示意图。如图 23(d)所示,第六比特集合中比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 中每个比特方阵中的一半比特(128 比特)来自第二比特集合,另外一半比特(128 比特)来自未经过 PCS 处理的第三比特集合。比特方阵 $I_{0,5}$ 和 $I_{1,5}$ 各自每行有 9 个比特来自经过 PCS 处理的第二比特集合,每行中另外 7 个比特来自未经过 PCS 处理的第三比特集合。比特方阵 $I_{0,6}$ 和 $I_{1,6}$ 中所有比特来自经过 PCS 处理的第二比特集合。

图 11(e)为本申请实施例中多个第一比特子集的另一种分布示意图。基于图 23(d)所示的第六比特集合经过 FEC 编码得到第四比特集合,则对应的第四比特集合分布如图 11(e)所示。具体地,第四比特集合中的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 中每个比特方阵中的一半比特(128 比特)来自第二比特集合,另外一半比特(128 比特)来自未经过 PCS 处理的第三比特集合。比特方阵 $B_{0,5}$ 和 $B_{1,5}$ 各自每行有 9 个比特来自经过 PCS 处理的第二比特集合,每行中另外的 7 个比特来自未经过 PCS 处理的第三比特集合。比特方阵 $B_{0,6}$ 和 $B_{1,6}$ 各自的第 i_1 行第 $15 - i_1$ 列的比特来自 OFEC 校验比特, $0 \leq i_1 \leq 15$, 即比特方阵 $B_{0,6}$ 和 $B_{1,6}$ 中位于副对角线上的比特来自 OFEC 校验比特,而其他比特来自经过 PCS 处理的第二比特集合。比特方阵 $B_{0,7}$ 和 $B_{1,7}$ 的所有比特来自 OFEC 校验比特。

进一步,图 11(e)所示的第四比特集合中的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 中来自第二比特集合的比特和来自第三比特集合的比特排列方式如图 9(a)或者图 9(b)所示。图 11(e)所示的比特方阵 $B_{0,5}$ 和 $B_{1,5}$ 各自第 0 列到第 C-1 列中来自第二比特集合的比特和来自第三比特集合的比特排列方式如图 9(a)的第 0 列到第 C-1 列或者如图 9(b)的第 0 列到第 C-1 列所示。在此情况

下对第四比特集合中 $B_{0,5}$ 的第C列到第15列、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,5}$ 的第C列到第15列、 $B_{1,6}$ 和 $B_{1,7}$ 进行第一交织得到第五比特集合中 $T_{0,5}$ 的第C列到第15列、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,5}$ 的第C列到第15列、 $T_{1,6}$ 和 $T_{1,7}$ ，其中， $0 \leq C \leq 15$ 。下面介绍几种具体的第一交织方案。

在一种可能的实施方式中，基于图23(d)所示的第六比特集合，若第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 如图24(a)所示，则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 如图9(a)所示。若第六比特集合的比特方阵 $I_{0,5}$ 和 $I_{1,5}$ 如图29(a)所示，则第四比特集合中的比特方阵 $B_{0,5}$ 、 $B_{1,5}$ 如图30(a)所示。图29(a)中标记的4个阴影部分均为 4×4 比特方阵，每个 4×4 比特方阵中每行的其中3个比特来自经过PCS处理的第二比特集合，每行剩余的1个比特来自未经过PCS处理的第三比特集合。图30(a)中标记的阴影部分为第12列至第15列的16行4列的比特方阵，在该16行4列的比特方阵中每行的3个比特来自经过PCS处理的第二比特集合，每行的剩余1个比特来自未经过PCS处理的第三比特集合。

进一步，若第六比特集合的比特方阵 $I_{0,5}$ 和 $I_{1,5}$ 如图29(b)、图29(c)、或图29(d)或图29(e)所示，则第四比特集合中的比特方阵 $B_{0,5}$ 和 $B_{1,5}$ 依次对应如图30(b)、图30(c)、图30(d)或图30(e)所示。

在上述情景下，对第四比特集合中 $B_{0,5}$ 的第12列到第15列、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,5}$ 的第12列到第15列、 $B_{1,6}$ 和 $B_{1,7}$ 进行第一交织得到第五比特集合中 $T_{0,5}$ 的第12列到第15列、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,5}$ 的第12列到第15列、 $T_{1,6}$ 和 $T_{1,7}$ 。而第四比特集合中 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 和 $B_{1,4}$ 按照原有的顺序输出得到第五比特集合中 $T_{0,0}$ 、 $T_{0,1}$ 、 $T_{0,2}$ 、 $T_{0,3}$ 、 $T_{0,4}$ 、 $T_{1,0}$ 、 $T_{1,1}$ 、 $T_{1,2}$ 、 $T_{1,3}$ 和 $T_{1,4}$ 。第四比特集合中比特方阵 $B_{0,5}$ 的第0列到第11列以及比特方阵 $B_{1,5}$ 的第0列到第11列按照原有的顺序输出得到第五比特集合中的比特方阵 $T_{0,5}$ 的第0列到第11列以及比特方阵 $T_{1,5}$ 的第0列到第11列。经过第一交织后第五比特集合的每个比特方阵如图9(a)所示，即第五比特集合的每个比特方阵中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + i_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特来自第二比特集合， $0 \leq i_2 < 16$ ， $0 \leq j_2 < 8$ 。此时对应的第一交织的一种具体交织规则如下：

包括 m_0 个比特方阵T的第五比特集合中第 r_1 行的128个比特中的第92比特到第127比特来自包括 m_0 个比特方阵B的第四比特集合中第 r_0 行的第92比特到第127中的比特。除此之外，第一交织的具体交织规则还有很多种，此处不再逐一列举。

在另一种可能的实施方式中，基于图23(d)所示的第六比特集合，若第六比特集合的比特方阵 $I_{0,0}$ 、 $I_{0,1}$ 、 $I_{0,2}$ 、 $I_{0,3}$ 、 $I_{0,4}$ 、 $I_{1,0}$ 、 $I_{1,1}$ 、 $I_{1,2}$ 、 $I_{1,3}$ 、 $I_{1,4}$ 如图24(b)所示，则第四比特集合的比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 如图9(b)所示。若第六比特集合的比特方阵 $I_{0,5}$ 和 $I_{1,5}$ 如图29(f)所示，则第四比特集合中的比特方阵 $B_{0,5}$ 和 $B_{1,5}$ 如图30(f)所示。在此情景下，对第四比特集合中 $B_{0,5}$ 的第14列至第15列、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,5}$ 的第14列至第15列、 $B_{1,6}$ 和 $B_{1,7}$ 进行第一交织得到第五比特集合中 $T_{0,5}$ 的第14列至第15列、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,5}$ 的第14列至第15列、 $T_{1,6}$ 和 $T_{1,7}$ 。而第四比特集合中 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 和 $B_{1,4}$ 按照原有的顺序输出得到第五比特集合中的比特方阵 $T_{0,0}$ 、 $T_{0,1}$ 、 $T_{0,2}$ 、 $T_{0,3}$ 、 $T_{0,4}$ 、 $T_{1,0}$ 、 $T_{1,1}$ 、 $T_{1,2}$ 、 $T_{1,3}$ 和 $T_{1,4}$ 。第四比特集合中比特方阵 $B_{0,5}$ 的第0列至第13列以及比特方阵 $B_{1,5}$ 的第0列至第13列按照原有的顺序输出得到第五比特集合中的特方阵 $T_{0,5}$ 的第0列至第13列以及比特方阵 $T_{1,5}$ 的第0列至第13列。经过第一交织后第五比特集合的每个比特方阵如图9(b)所示，即其每个比特方阵的第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自第二比特集合， $0 \leq i_2 < 16$ ， $0 \leq j_2 < 8$ 。此时对应的第一交织的一种具体交织规则如下：包括 m_0 个比特方阵T的第五比特集合中第 r_1 行的128个比特中的第94比特到第127比特来自包括 m_0 个比特方阵B第四比特集合中第 r_0 行的第94比特到第127中的比特。除此之外，第一交织的具体交织规则还有很多种，此处不再逐一列举。

需要说明的是，在一些具体场景中，可以满足 $r_0 = r_1$ 。

图12为本申请实施例中第一交织处理的第二种实施方式示意图。如图12所示，在一种可能的实施方式中，第二比特子集中每行16个比特来自2个第一比特子集。更进一步，第二比特子集中每行16个比特的其中8个比特来自其中一个第一比特子集的其中一行，第二比特子集中每行16个比特的另外8个比特来自另外一个第一比特子集的其中一行。

图 13 为本申请实施例中第一交织处理的第三种实施方式示意图。如图 13 所示，在一种可能的实施方式中，第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列，第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列，第五比特集合中每列 32 个比特来自 2 个第一比特子集。更具体地，第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的其中一列，第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的其中一列。进一步，第五比特集合中每列 32 个比特的其中 16 个比特来自其中一个第一比特子集的第 $c_1 \% 16$ 列，第五比特集合中每列 32 个比特的另外 16 个比特来自另外一个第一比特子集的第 $c_1 \% 16$ 列，其中， $0 \leq c_1 < (m_0 \times 8)$ 。

(3) 第二交织处理：

具体以第二交织处理采用 OFEC 交织为例，OFEC 交织器接收 2 个第一交织处理器输出的比特，进行“方阵内交织”（intra-block interleaving）和“方阵间交织”（inter-block interleaving）。

图 14 为本申请实施例中方阵内交织的一个示意图。如图 14 所示，OFEC 交织器先将每个接收的方阵进行“方阵内交织”（intra-block interleaving），也就是每个 16 行 16 列的输入方阵按如图 14 所示的交织规则分别进行交织打乱顺序得到一个 16 行 16 列的输出方阵。图 14 中第 i_1 行第 j_1 列（ $0 \leq i_1 < 16$ 且 $0 \leq j_1 < 16$ ）的元素为 (a, b) ，则表明经过“方阵内交织”后的输出方阵中第 i_1 行第 j_1 列的比特来自输入方阵中第 a 行第 b 列的比特。例如，图 14 中第 1 行第 0 列的元素为 $(14, 15)$ ，则经过“方阵内交织”后的输出方阵中第 1 行第 0 列的比特来自输入方阵中第 14 行第 15 列的比特。

经过“方阵内交织”处理后的比特进行“方阵间交织”（inter-block interleaving），以提升整体抗突发性能。下面对方阵间交织的操作进行描述。

图 15 为本申请实施例中方阵间交织的一个示意图。如图 15 所示，方阵间交织包含 84 行 8 列的交织缓存（interleaver buffer） M ，每行包含 8 个方阵，每个方阵包含 16 行 16 列总共 256 个比特。方阵间交织的交织缓存大小为 $84 \times 8 \times 256 = 172032$ 比特，对应 $172032/4096=42$ 个第一交织处理器的输出分组，其中 21 个输出分组来自其中一个第一交织处理器的输出分组且位于交织缓存 M 中的偶数行，另外 21 个输出分组来自另一个第一交织处理器的输出分组且位于交织缓存 M 中的奇数行。这里每个第一交织处理器的输出分组为 2 行 8 列方阵集合总共 4096 个比特。方阵间交织的交织缓存 M 可划分为 4 个集合，如图 15 所示，第 0 集合包含交织缓存 M 中第 0, 2, 4, ..., 40 行的方阵总共 $21 \times 16 \times 128 = 43008$ 个比特，第 1 集合包含交织缓存 M 中第 1, 3, 5, ..., 41 行的方阵总共 43008 个比特，第 2 集合包含交织缓存 M 中第 42, 44, 46, ..., 82 行的方阵总共 43008 个比特，第 3 集合包含交织缓存 M 中第 43, 45, 47, ..., 83 行的方阵总共 43008 个比特。

从交织缓存 M 中以 8 比特为颗粒度，轮询从各个集合中读取每列中的比特，将其每列中的所有比特读出后再读取下一列中的比特。先从第 0 集合中从上到下读取最开始的一组 8 个比特，接着从第 1 集合、第 2 集合、第 3 集合中分别从上到下读取最开始的 8 个比特，一个循环（cycle）总共读取 32 比特。接着，下一个循环从第 0 集合、第 1 集合、第 2 集合、第 3 集合中分别从上到下读取下一组 8 个比特，总共读取 32 比特。经过总共 42 个循环，将当前 1 列 1344 个比特读出。对于从各个集合中读取每列中的比特，具体操作如下：

先从交织缓存 M 第 0 行方阵中从上到下依次读取 8 比特（方阵中第 0-7 比特行），
从交织缓存 M 中第 1 行方阵中从上到下依次读取 8 比特（方阵中第 0-7 比特行），
从交织缓存 M 中第 42 行方阵中从上到下依次读取 8 比特（方阵中第 0-7 比特行），
从交织缓存 M 中第 43 行方阵中从上到下依次读取 8 比特（方阵中第 0-7 比特行），
然后从交织缓存 M 中第 0 行方阵从上到下依次读取 8 比特（方阵中第 8-15 比特行），
从交织缓存 M 中第 1 行方阵从上到下依次读取 8 比特（方阵中第 8-15 比特行），
从交织缓存 M 中第 42 行方阵从上到下依次读取 8 比特（方阵中第 8-15 比特行），
从交织缓存 M 中第 43 行方阵从上到下依次读取 8 比特（方阵中第 8-15 比特行），
再从交织缓存 M 中第 2 行方阵从上到下依次读取 8 比特（方阵中第 0-7 比特行），
...，直到将交织缓存 M 中当前比特列中总共 1344 个比特完全读出。

(4) 符号映射和偏振划分:

OFEC 交织输出的比特数据送入符号映射和偏振划分得到双偏振符号。如图 4 所示, 两个 OFEC 交织器, OFEC 交织 0 和 OFEC 交织 1, 以 8 比特为一组, 轮询送入符号映射。对于采用双偏振 16QAM 调制 (DP-16QAM) 下, 符号映射和偏振划分将每 8 个比特 ($b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$) 映射得到 1 个 DP-16QAM 符号。考虑如下的第一符号映射方案: (b_0, b_2) 映射为 DP-16QAM 符号在 X 偏振上 I 方向分部 (in-phase component), 记为 XI; (b_4, b_6) 映射为 DP-16QAM 符号在 X 偏振上 Q 方向分部 (quadrature-phase component), 记为 XQ; (b_1, b_3) 映射为 DP-16QAM 符号在 Y 偏振上 I 方向分部 (in-phase component), 记为 YI; (b_5, b_7) 映射为 DP-16QAM 符号在 Y 偏振上 Q 方向分部 (quadrature-phase component), 记为 YQ。对于每个信号维度 (signaling dimension) XI/XQ/YI/YQ, 将 2 个比特映射为对应符号幅度 (symbol amplitude) 采用如下映射方式:

$$(0,0) \rightarrow -3, (0,1) \rightarrow -1, (1,1) \rightarrow +1, (1,0) \rightarrow +3$$

此时, 8 个比特 ($b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$) 中的 b_0 和 b_4 为 X 偏振方向上 16QAM 中的 2 个符号位 (sign bits), b_2 和 b_6 为 X 偏振方向上 16QAM 中的 2 个幅度位 (amplitude bits); b_1 和 b_5 为 Y 偏振方向上 16QAM 中的 2 个符号位 (sign bits), b_3 和 b_7 为 Y 偏振方向上 16QAM 中的 2 个幅度位 (amplitude bits)。

需要说明的是, 采用如图 9 (a) 所示的第一交织处理, 其输出的 2 行 8 列比特方阵集合 T 中任一包含 256 个比特的方阵 T_{i_0, j_0} 中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特 (即图 9 (a) 中阴影背景的比特) 是来自经过 PCS 处理得到的 2048 个第三比特中的 1 个比特, 其中 $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

图 16 为本申请实施例中方阵内交织的另一个示意图。如图 16 所示, 给出了方阵 T_{i_0, j_0} 经过 “方阵内交织” 后得到的方阵内交织后方阵。第一交织处理输出的任一个 16 行 16 列的方阵经过 OFEC 交织中的 “方阵内交织” 后, 输出的任一个 16 行 16 列的方阵中, 第 2 行、第 3 行、第 6 行、第 7 行、第 10 行、第 11 行、第 14 行、第 15 行的每行 16 个比特均来自是来自经过 PCS 处理得到的 2048 个第三比特中的 16 个比特。

由于 “方阵间交织” 并不打乱方阵内的比特顺序, 经过 OFEC 交织中的 “方阵间交织” 后, 输出的任一个 16 行 16 列的方阵中, 第 2 行、第 3 行、第 6 行、第 7 行、第 10 行、第 11 行、第 14 行或第 15 行的 16 个比特均来自是来自经过 PCS 处理得到的 2048 个第三比特中的 16 个比特。进一步, 采用上述第一符号映射和偏振划分, 可在保持星座点位置不变情况下改变星座点出现的概率, 使其非均匀分布, 从而实现 PCS 以提升系统传输性能。

需要说明的是, 在一些具体实施例中, 所述符号映射和偏振划分将每 8 个比特 ($b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$) 映射得到 1 个 DP-16QAM 符号, 考虑如下的第二符号映射方案: (b_0, b_1) 映射为 DP-16QAM 符号在 X 偏振上 I 方向分部 (in-phase component), 记为 XI; (b_2, b_3) 映射为 DP-16QAM 符号在 X 偏振上 Q 方向分部 (quadrature-phase component), 记为 XQ; (b_4, b_5) 映射为 DP-16QAM 符号在 Y 偏振上 I 方向分部 (in-phase component), 记为 YI; (b_6, b_7) 映射为 DP-16QAM 符号在 Y 偏振上 Q 方向分部 (quadrature-phase component), 记为 YQ。对于每个信号维度 (signaling dimension) XI/XQ/YI/YQ, 将 2 个比特映射为对应符号幅度 (symbol amplitude) 采用如下映射方式:

$$(0,0) \rightarrow -3, (0,1) \rightarrow -1, (1,1) \rightarrow +1, (1,0) \rightarrow +3$$

此时, 8 个比特 ($b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$) 中的 b_0 和 b_2 为 X 偏振方向上 16QAM 中的 2 个符号位 (sign bits), b_1 和 b_3 为 X 偏振方向上 16QAM 中的 2 个幅度位 (amplitude bits); b_4 和 b_6 为 Y 偏振方向上 16QAM 中的 2 个符号位 (sign bits), b_5 和 b_7 为 Y 偏振方向上 16QAM 中的 2 个幅度位 (amplitude bits)。

需要说明的是, 采用如图 9 (b) 所示的第一交织处理, 其输出的 2 行 8 列比特方阵集合 T 中任一包含 256 个比特的方阵 T_{i_0, j_0} 中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特 (即图 9 (b) 中阴影背景的比特) 是来自经过 PCS 处理得到的 2048 个第三比特中的 1 个比特, 其中 $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。方阵 T_{i_0, j_0} 经过 “方阵内交织” 后得到的方阵内交织后方阵。第一交织处理输出的任一个 16 行 16 列的方阵经过 OFEC 交织中的 “方阵内交织” 后, 输出的任一个 16 行 16 列的方阵中, 第 1 行、第 3 行、第 5 行、第 7 行、第 9 行、第 11 行、第 13 行、第 15 行的每行 16 个比特均来自是来自经过 PCS 处理得到的 2048 个第三

比特中的 16 个比特。由于“方阵间交织”并不打乱方阵内的比特顺序，经过 OFEC 交织中的“方阵间交织”后，输出的任一个 16 行 16 列的方阵中，第 1 行、第 3 行、第 5 行、第 7 行、第 9 行、第 11 行、第 13 行、第 15 行的 16 个比特均来自是来自经过 PCS 处理得到的 2048 个第三比特中的 16 个比特。进一步，采用上述第二符号映射和偏振划分，可在保持星座点位置不变情况下改变星座点出现的概率，使其非均匀分布，从而实现 PCS 以提升系统传输性能。

(5) DSP 成帧：

经符号映射和偏振划分的双偏振符号流分别对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS 进行 DSP 成帧，得到待发送的双偏振符号流。

典型的 DSP 成帧将每 172032 个双偏振符号进行成帧得到 175104 个双偏振符号，175104 个双偏振符号也称为一个超帧，其冗余为 $\frac{1}{56}$ 。

需要说明的是，在一些具体应用中，上述“OFEC 交织输出的比特数据送入符号映射和偏振划分得到双偏振符号；经符号映射和偏振划分的双偏振符号流分别对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS 进行 DSP 成帧，得到待发送的双偏振符号流”可采用如下等效操作：先将 OFEC 交织输出的比特数据插入 FAW 帧对齐符号、TS 训练符号、RES 保留符号和 PS 导频符号在调制映射前所对应的比特序列，然后再进行符号映射和偏振划分，得到相同的待发送双偏振符号流。其具体操作在本申请中，不再进行具体展开。

需要说明的是，给定数据帧的速率为 $W_{\text{bit}} \text{ Gbit/s}$ ，经过第一数据处理对从数据帧中获取的 r 行数据进行 CRC 校验，和/或插入填充 (Pad) 比特，再进行加扰 (scrambling)，得到 $10280 \times r_{\text{Frame}} + d_{\text{CRC}+\text{PAD}}$ 个加扰比特，其对应的加扰后速率为 $W \times (10280 \times r_{\text{Frame}} + d_{\text{CP}}) / (10280 \times r_{\text{Frame}}) \text{ Gbit/s}$ 。加扰后的比特数据进行包含 PCS 和 OFEC 编码、第一交织处理、OFEC 交织、16QAM 符号映射和偏振划分，并进行冗余为 $\frac{1}{56}$ 的 DSP 成帧的第二数据处理后，得到的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times (10280 \times r_{\text{Frame}} + d_{\text{CP}}) / (10280 \times r_{\text{Frame}}) \times (4096/k) \times (57/56) / 8 \text{ Gbaud}$ 。表 1 给出了多种数据帧类型及其对应标称速率 (nominal bit rate)。

表 1

数据帧类型	标称速率 Gbit/s	OTN/Ethernet
FlexO-6e	~603.7346	Ethernet
FlexO-6	$99.5328 \times 6 \times \frac{239}{226} \times \frac{4112}{4097} \approx 633.8611$	OTN
FlexO-8e	~804.9795	Ethernet
FlexO-8	$99.5328 \times 8 \times \frac{239}{226} \times \frac{4112}{4097} \approx 845.1481$	OTN

由于当前产业界典型的低功耗器件对应的最高波特率约为 131Gbaud。这里考虑 W_{baud} 小于 132Gbaud 的情况。表 2 和表 3 给出了多种参数组合，包括每次从数据帧中获取数据行数 r 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 、采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下对应的波特率。其中，表 2 给出了的任一参数组合，采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下对应的波特率至少有一种数据帧类型下对应的波特率是小于 132Gbaud 的，而且 CRC 校验和插入填充比特对应的冗余 OH_{CP} 不大于 0.1% 使其不影响整体系统传输效率。

第一数据处理输出的加扰比特的数量 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CP}}$ 是 $4 \times k$ 的整数倍，使得整个数据处理的具体硬件实现较简单，功耗较低；PCS 处理对应的输入比特数量 $k_{\text{PCS}} = k - 1504$ 是 2 的整数倍，使得 PCS 处理便于硬件实现且具有较优性能。同时第一数据处理输出的 d_{scr} 个加扰比特经 PCS 处理和 OFEC

编码后得到的 $d_{scr}/k \times 4096$ 个 OFEC 编码比特, $d_{scr}/k \times 4096$ 是 2 个 OFEC 交织器总大小 $172032 \times 2 = 344064$ 的整数倍, 使得整个数据处理的具体硬件实现较简单, 功耗较低。

需要说明的, 给定一组参数组合, 即下表 2 或表 3 中的一行中的 r_{Frame} 、 d_{CP} 、 k , 其对应的采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下的波特率可能部分大于 132Gbaud, 则表明采用该组参数组合不适用该数据帧类型, 其在表格中采用背景阴影标注。需要说明的, 对于给定一组参数组合, 即下表 2 中的一行中的 r_{Frame} 、 d_{CP} 、 k , 其对应的采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下的波特率可能部分小于 110Gbaud, 则表明采用该组参数组合也不适用该数据帧类型, 其在表格中采用背景阴影标注。例如, 对于序号为 1 的参数组合 $\{r_{Frame} = 79, d_{CP} = 328, k = 3224\}$, 采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下的波特率分别为 97.6Gbaud、102.5Gbaud、123.9Gbaud、130.1Gbaud, 其中 FlexO-6e、FlexO-6 数据帧类型下的波特率小于 110Gbaud, 表明该参数组合不适用 FlexO-6e、FlexO-6 数据帧类型使用, 其在表格中采用背景阴影标注; FlexO-8 数据帧类型下的波特率大于 132Gbaud, 表明该组参数不适用 FlexO-8 数据帧类型使用, 其在表格中采用背景阴影标注; FlexO-8e 数据帧类型下的波特率大于 110Gbaud 且小于 132Gbaud, 表明该组参数适用 FlexO-8e 数据帧类型使用。又例如, 对于表 2 中序号为 4 的参数组合 $\{r_{Frame} = 87, d_{CP} = 72, k = 2662\}$, 采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下的波特率分别为 118.2Gbaud、124.1Gbaud、157.6Gbaud、165.5Gbaud, 其中 FlexO-8e、FlexO-8 数据帧类型下的波特率大于 132Gbaud, 表明该组参数不适用 FlexO-8e、FlexO-8 数据帧类型使用, 其在表格中采用背景阴影标注; FlexO-6e、FlexO-6 数据帧类型下的波特率大于 110Gbaud 且小于 132Gbaud, 表明该组参数适用 FlexO-6e、FlexO-6 数据帧类型使用。

表 2

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud			
						FlexO-6e	FlexO-6	FlexO-8e	FlexO-8
1	79	328	3224	1720	0.040%	97.6	102.5	130.2	136.7
2	83	32	3386	1882	0.004%	92.9	97.6	123.9	130.1
3	83	536	3388	1884	0.063%	92.9	97.6	123.9	130.1
4	87	72	2662	1158	0.008%	118.2	124.1	157.6	165.5
5	87	744	2664	1160	0.083%	118.2	124.1	157.6	165.5
6	105	504	3214	1710	0.047%	97.9	102.8	130.6	137.1
7	109	40	2668	1164	0.004%	117.9	123.8	157.2	165.1
8	109	880	2670	1166	0.079%	117.9	123.8	157.2	165.1
9	110	176	3366	1862	0.016%	93.5	98.2	124.7	130.9
10	110	848	3368	1864	0.075%	93.5	98.2	124.7	130.9
11	111	648	3398	1894	0.057%	92.6	97.3	123.5	129.7
12	131	8	2672	1168	0.001%	117.8	123.6	157.0	164.8
13	131	1016	2674	1170	0.075%	117.8	123.6	157.0	164.8
14	131	680	3208	1704	0.051%	98.1	103.0	130.8	137.4
15	132	480	3232	1728	0.035%	97.4	102.2	129.8	136.3
16	132	1320	3234	1730	0.097%	97.4	102.2	129.8	136.3
17	137	320	3354	1850	0.023%	93.8	98.5	125.1	131.3
18	137	1160	3356	1852	0.082%	93.8	98.5	125.1	131.3
19	138	120	3378	1874	0.009%	93.1	97.8	124.2	130.4
20	138	960	3380	1876	0.068%	93.1	97.8	124.2	130.4

21	152	344	2658	1154	0.022%	118.4	124.3	157.9	165.7
22	152	1520	2660	1156	0.097%	118.4	124.3	157.9	165.7
23	153	648	2676	1172	0.041%	117.6	123.5	156.8	164.7
24	157	856	3204	1700	0.053%	98.3	103.2	131.0	137.5
25	158	656	3224	1720	0.040%	97.6	102.5	130.2	136.7
26	165	264	3366	1862	0.016%	93.5	98.2	124.7	130.9
27	165	1272	3368	1864	0.075%	93.5	98.2	124.7	130.9
28	166	64	3386	1882	0.004%	92.9	97.6	123.9	130.1
29	166	1072	3388	1884	0.063%	92.9	97.6	123.9	130.1
30	174	144	2662	1158	0.008%	118.2	124.1	157.6	165.5
31	174	1488	2664	1160	0.083%	118.2	124.1	157.6	165.5
32	175	616	2678	1174	0.034%	117.5	123.4	156.7	164.5
33	183	360	3200	1696	0.019%	98.3	103.2	131.1	137.7
34	183	1536	3202	1698	0.082%	98.3	103.2	131.1	137.7
35	184	664	3218	1714	0.035%	97.8	102.7	130.4	136.9
36	184	1840	3220	1716	0.097%	97.8	102.7	130.4	136.9
37	185	968	3236	1732	0.051%	97.3	102.1	129.7	136.2
38	192	744	3358	1854	0.038%	93.7	98.4	125.0	131.2
39	192	1920	3360	1856	0.097%	93.7	98.4	125.0	131.2
40	193	1048	3376	1872	0.053%	93.2	97.9	124.3	130.5
41	194	176	3392	1888	0.009%	92.8	97.4	123.7	129.9
42	194	1352	3394	1890	0.068%	92.8	97.4	123.7	129.9
43	196	616	2666	1162	0.031%	118.1	123.9	157.4	165.3
44	209	536	3198	1694	0.025%	98.4	103.3	131.2	137.8
45	209	1880	3200	1696	0.088%	98.4	103.3	131.2	137.8
46	210	1008	3214	1710	0.047%	97.9	102.8	130.6	137.1
47	211	136	3228	1724	0.006%	97.5	102.3	130.0	136.5
48	211	1480	3230	1726	0.068%	97.5	102.3	130.0	136.5
49	217	280	2656	1152	0.013%	118.5	124.4	158.0	165.9
50	217	1960	2658	1154	0.088%	118.5	124.4	158.0	165.9
51	218	80	2668	1164	0.004%	117.9	123.8	157.2	165.1
52	218	1760	2670	1166	0.079%	117.9	123.8	157.2	165.1
53	219	1224	3352	1848	0.054%	93.9	98.6	125.2	131.5
54	220	352	3366	1862	0.016%	93.5	98.2	124.7	130.9
55	220	1696	3368	1864	0.075%	93.5	98.2	124.7	130.9
56	221	824	3382	1878	0.036%	93.1	97.7	124.1	130.3
57	221	2168	3384	1880	0.095%	93.1	97.7	124.1	130.3
58	222	1296	3398	1894	0.057%	92.6	97.3	123.5	129.7
59	235	376	3196	1692	0.016%	98.5	103.4	131.3	137.8

60	235	1888	3198	1694	0.078%	98.5	103.4	131.3	137.8
61	236	680	3210	1706	0.028%	98.0	102.9	130.7	137.2
62	236	2192	3212	1708	0.090%	98.0	102.9	130.7	137.2
63	237	984	3224	1720	0.040%	97.6	102.5	130.2	136.7
64	238	1288	3238	1734	0.053%	97.2	102.1	129.6	136.1
65	239	920	2660	1156	0.037%	118.3	124.2	157.8	165.6
66	240	1728	2672	1168	0.070%	117.8	123.7	157.1	165.0
67	247	1000	3360	1856	0.039%	93.7	98.4	124.9	131.1
68	247	2512	3362	1858	0.099%	93.7	98.4	124.9	131.1
69	248	1304	3374	1870	0.051%	93.3	98.0	124.4	130.6
70	249	96	3386	1882	0.004%	92.9	97.6	123.9	130.1
71	249	1608	3388	1884	0.063%	92.9	97.6	123.9	130.1
72	250	400	3400	1896	0.016%	92.6	97.2	123.4	129.6
73	250	1912	3402	1898	0.074%	92.6	97.2	123.4	129.6

表 3

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud			
						FlexO-6e	FlexO-6	FlexO-8e	FlexO-8
1	261	216	2662	1158	0.008%	118.203	124.102	157.604	165.469
2	261	2232	2664	1160	0.083%	118.203	124.102	157.604	165.469
3	261	1560	3196	1692	0.058%	98.503	103.418	131.337	137.891
4	262	16	2672	1168	0.001%	117.752	123.628	157.003	164.837
5	262	2032	2674	1170	0.075%	117.752	123.628	157.003	164.837
6	262	1360	3208	1704	0.051%	98.127	103.023	130.836	137.364
7	263	1160	3220	1716	0.043%	97.754	102.632	130.338	136.842
8	264	960	3232	1728	0.035%	97.383	102.243	129.845	136.324
9	264	2640	3234	1730	0.097%	97.383	102.243	129.845	136.324
10	274	640	3354	1850	0.023%	93.829	98.511	125.106	131.349
11	274	2320	3356	1852	0.082%	93.829	98.511	125.106	131.349
12	275	440	3366	1862	0.016%	93.488	98.153	124.651	130.871
13	275	2120	3368	1864	0.075%	93.488	98.153	124.651	130.871
14	276	240	3378	1874	0.009%	93.149	97.798	124.199	130.397
15	276	1920	3380	1876	0.068%	93.149	97.798	124.199	130.397
16	277	40	3390	1886	0.001%	92.813	97.444	123.751	129.926
17	277	1720	3392	1888	0.060%	92.813	97.444	123.751	129.926
18	283	2032	2666	1162	0.070%	118.099	123.992	157.465	165.323
19	284	488	2674	1170	0.017%	117.683	123.555	156.911	164.741
20	284	2672	2676	1172	0.092%	117.683	123.555	156.911	164.741
21	287	896	3194	1690	0.030%	98.537	103.454	131.383	137.939
22	287	2744	3196	1692	0.093%	98.537	103.454	131.383	137.939
23	288	1704	3206	1702	0.058%	98.195	103.095	130.927	137.460
24	289	664	3216	1712	0.022%	97.855	102.738	130.474	136.984

25	289	2512	3218	1714	0.085%	97.855	102.738	130.474	136.984
26	290	1472	3228	1724	0.049%	97.518	102.384	130.024	136.512
27	291	432	3238	1734	0.014%	97.183	102.032	129.577	136.043
28	291	2280	3240	1736	0.076%	97.183	102.032	129.577	136.043
29	302	80	3360	1856	0.003%	93.643	98.316	124.857	131.088
30	302	1928	3362	1858	0.062%	93.643	98.316	124.857	131.088
31	303	888	3372	1868	0.029%	93.334	97.991	124.445	130.655
32	303	2736	3374	1870	0.088%	93.334	97.991	124.445	130.655
33	304	688	2658	1154	0.022%	118.398	124.306	157.864	165.741
34	304	3040	2660	1156	0.097%	118.398	124.306	157.864	165.741
35	304	1696	3384	1880	0.054%	93.027	97.669	124.036	130.225
36	305	2168	2668	1164	0.069%	118.010	123.898	157.346	165.198
37	305	656	3394	1890	0.021%	92.722	97.349	123.629	129.798
38	305	2504	3396	1892	0.080%	92.722	97.349	123.629	129.798
39	306	1296	2676	1172	0.041%	117.624	123.493	156.832	164.658
40	313	1912	3194	1690	0.059%	98.566	103.484	131.421	137.979
41	314	1712	3204	1700	0.053%	98.252	103.155	131.002	137.539
42	315	1512	3214	1710	0.047%	97.940	102.827	130.587	137.103
43	316	1312	3224	1720	0.040%	97.630	102.502	130.173	136.669
44	317	1112	3234	1730	0.034%	97.322	102.178	129.763	136.238
45	317	3128	3236	1732	0.096%	97.322	102.178	129.763	136.238
46	326	320	2660	1156	0.010%	118.294	124.197	157.725	165.596
47	326	2840	2662	1158	0.085%	118.294	124.197	157.725	165.596
48	327	120	2668	1164	0.004%	117.932	123.817	157.243	165.089
49	327	2640	2670	1166	0.079%	117.932	123.817	157.243	165.089
50	328	2440	2678	1174	0.072%	117.573	123.440	156.764	164.586
51	329	728	3356	1852	0.022%	93.772	98.452	125.030	131.269
52	329	2744	3358	1854	0.081%	93.772	98.452	125.030	131.269
53	330	528	3366	1862	0.016%	93.488	98.153	124.651	130.871
54	330	2544	3368	1864	0.075%	93.488	98.153	124.651	130.871
55	331	328	3376	1872	0.010%	93.206	97.857	124.274	130.476
56	331	2344	3378	1874	0.069%	93.206	97.857	124.274	130.476
57	332	128	3386	1882	0.004%	92.925	97.562	123.900	130.083
58	332	2144	3388	1884	0.063%	92.925	97.562	123.900	130.083
59	333	1944	3398	1894	0.057%	92.646	97.269	123.528	129.692
60	339	744	3192	1688	0.021%	98.590	103.510	131.453	138.013
61	339	2928	3194	1690	0.084%	98.590	103.510	131.453	138.013
62	340	1384	3202	1698	0.040%	98.300	103.205	131.067	137.607
63	341	2024	3212	1708	0.058%	98.012	102.903	130.682	137.203
64	342	480	3220	1716	0.014%	97.725	102.602	130.300	136.802
65	342	2664	3222	1718	0.076%	97.725	102.602	130.300	136.802
66	343	1120	3230	1726	0.032%	97.440	102.303	129.920	136.403
67	343	3304	3232	1728	0.094%	97.440	102.303	129.920	136.403

68	344	1760	3240	1736	0.050%	97.157	102.005	129.543	136.007
69	348	288	2662	1158	0.008%	118.203	124.102	157.604	165.469
70	348	2976	2664	1160	0.083%	118.203	124.102	157.604	165.469
71	349	760	2670	1166	0.021%	117.865	123.746	157.153	164.995
72	349	3448	2672	1168	0.096%	117.865	123.746	157.153	164.995
73	350	1232	2678	1174	0.034%	117.528	123.393	156.704	164.523
74	356	704	3352	1848	0.019%	93.882	98.567	125.176	131.422
75	356	2888	3354	1850	0.079%	93.882	98.567	125.176	131.422
76	357	1344	3362	1858	0.037%	93.619	98.291	124.825	131.054
77	357	3528	3364	1860	0.096%	93.619	98.291	124.825	131.054
78	358	1984	3372	1868	0.054%	93.358	98.016	124.477	130.688
79	359	440	3380	1876	0.012%	93.097	97.743	124.130	130.324
80	359	2624	3382	1878	0.071%	93.097	97.743	124.130	130.324
81	360	1080	3390	1886	0.029%	92.839	97.472	123.785	129.962
82	360	3264	3392	1888	0.088%	92.839	97.472	123.785	129.962
83	361	1720	3400	1896	0.046%	92.582	97.202	123.442	129.602
84	365	1592	3192	1688	0.042%	98.611	103.531	131.481	138.042
85	366	720	3200	1696	0.019%	98.341	103.249	131.122	137.665
86	366	3072	3202	1698	0.082%	98.341	103.249	131.122	137.665
87	367	2200	3210	1706	0.058%	98.073	102.967	130.764	137.290
88	368	1328	3218	1714	0.035%	97.807	102.687	130.409	136.917
89	368	3680	3220	1716	0.097%	97.807	102.687	130.409	136.917
90	369	2304	2658	1154	0.061%	118.444	124.354	157.925	165.805
91	369	456	3226	1722	0.012%	97.542	102.409	130.056	136.546
92	369	2808	3228	1724	0.074%	97.542	102.409	130.056	136.546
93	370	592	2664	1160	0.016%	118.123	124.018	157.498	165.357
94	370	3448	2666	1162	0.091%	118.123	124.018	157.498	165.357
95	370	1936	3236	1732	0.051%	97.278	102.132	129.704	136.176
96	371	1736	2672	1168	0.046%	117.805	123.684	157.073	164.911
97	384	1488	3358	1854	0.038%	93.732	98.409	124.975	131.212
98	384	3840	3360	1856	0.097%	93.732	98.409	124.975	131.212
99	385	616	3366	1862	0.016%	93.488	98.153	124.651	130.871
100	385	2968	3368	1864	0.075%	93.488	98.153	124.651	130.871
101	386	2096	3376	1872	0.053%	93.246	97.899	124.328	130.532
102	387	1224	3384	1880	0.031%	93.005	97.646	124.007	130.195
103	387	3576	3386	1882	0.090%	93.005	97.646	124.007	130.195
104	388	352	3392	1888	0.009%	92.765	97.394	123.687	129.859
105	388	2704	3394	1890	0.068%	92.765	97.394	123.687	129.859
106	389	1832	3402	1898	0.046%	92.527	97.144	123.369	129.525
107	391	2440	2660	1156	0.061%	118.355	124.260	157.806	165.681
108	392	1232	2666	1162	0.031%	118.053	123.943	157.403	165.258
109	392	2240	3200	1696	0.056%	98.377	103.286	131.170	137.715
110	393	24	2672	1168	0.001%	117.752	123.628	157.003	164.837

111	393	3048	2674	1170	0.075%	117.752	123.628	157.003	164.837
112	393	2040	3208	1704	0.051%	98.127	103.023	130.836	137.364
113	394	1840	3216	1712	0.045%	97.878	102.762	130.504	137.016
114	395	1640	3224	1720	0.040%	97.630	102.502	130.173	136.669
115	396	1440	3232	1728	0.035%	97.383	102.243	129.845	136.324
116	396	3960	3234	1730	0.097%	97.383	102.243	129.845	136.324
117	397	1240	3240	1736	0.030%	97.138	101.985	129.518	135.980
118	397	3760	3242	1738	0.092%	97.138	101.985	129.518	135.980
119	411	960	3354	1850	0.023%	93.829	98.511	125.106	131.349
120	411	3480	3356	1852	0.082%	93.829	98.511	125.106	131.349
121	412	760	3362	1858	0.018%	93.602	98.272	124.802	131.030
122	412	3280	3364	1860	0.077%	93.602	98.272	124.802	131.030
123	413	2912	2662	1158	0.069%	118.275	124.177	157.700	165.569
124	413	560	3370	1866	0.013%	93.375	98.034	124.500	130.712
125	413	3080	3372	1868	0.073%	93.375	98.034	124.500	130.712
126	414	2208	2668	1164	0.052%	117.989	123.877	157.319	165.169
127	414	360	3378	1874	0.009%	93.149	97.798	124.199	130.397
128	414	2880	3380	1876	0.068%	93.149	97.798	124.199	130.397
129	415	1504	2674	1170	0.035%	117.705	123.578	156.940	164.771
130	415	160	3386	1882	0.004%	92.925	97.562	123.900	130.083
131	415	2680	3388	1884	0.063%	92.925	97.562	123.900	130.083
132	416	2480	3396	1892	0.058%	92.702	97.327	123.602	129.770
133	418	1072	3198	1694	0.025%	98.409	103.319	131.211	137.759
134	418	3760	3200	1696	0.088%	98.409	103.319	131.211	137.759
135	419	1544	3206	1702	0.036%	98.174	103.073	130.898	137.430
136	419	4232	3208	1704	0.098%	98.174	103.073	130.898	137.430
137	420	2016	3214	1710	0.047%	97.940	102.827	130.587	137.103
138	421	2488	3222	1718	0.058%	97.707	102.583	130.276	136.777
139	422	272	3228	1724	0.006%	97.476	102.340	129.968	136.453
140	422	2960	3230	1726	0.068%	97.476	102.340	129.968	136.453
141	423	744	3236	1732	0.017%	97.245	102.098	129.660	136.130
142	423	3432	3238	1734	0.079%	97.245	102.098	129.660	136.130
143	434	560	2656	1152	0.013%	118.476	124.388	157.968	165.850
144	434	3920	2658	1154	0.088%	118.476	124.388	157.968	165.850
145	435	360	2662	1158	0.008%	118.203	124.102	157.604	165.469
146	435	3720	2664	1160	0.083%	118.203	124.102	157.604	165.469
147	436	160	2668	1164	0.004%	117.932	123.817	157.243	165.089
148	436	3520	2670	1166	0.079%	117.932	123.817	157.243	165.089
149	437	3320	2676	1172	0.074%	117.662	123.534	156.883	164.712
150	438	2448	3352	1848	0.054%	93.915	98.601	125.220	131.468
151	439	232	3358	1854	0.005%	93.701	98.377	124.935	131.169
152	439	2920	3360	1856	0.065%	93.701	98.377	124.935	131.169
153	440	704	3366	1862	0.016%	93.488	98.153	124.651	130.871

154	440	3392	3368	1864	0.075%	93.488	98.153	124.651	130.871
155	441	1176	3374	1870	0.026%	93.276	97.931	124.368	130.574
156	441	3864	3376	1872	0.085%	93.276	97.931	124.368	130.574
157	442	1648	3382	1878	0.036%	93.065	97.709	124.087	130.279
158	442	4336	3384	1880	0.095%	93.065	97.709	124.087	130.279
159	443	2120	3390	1886	0.047%	92.855	97.488	123.807	129.985
160	444	2424	3198	1694	0.053%	98.436	103.348	131.248	137.798
161	444	2592	3398	1894	0.057%	92.646	97.269	123.528	129.692
162	445	712	3204	1700	0.016%	98.215	103.116	130.953	137.488
163	445	3568	3206	1702	0.078%	98.215	103.116	130.953	137.488
164	446	1856	3212	1708	0.041%	97.995	102.885	130.660	137.180
165	447	144	3218	1714	0.003%	97.776	102.655	130.367	136.873
166	447	3000	3220	1716	0.065%	97.776	102.655	130.367	136.873
167	448	1288	3226	1722	0.028%	97.557	102.425	130.076	136.567
168	448	4144	3228	1724	0.090%	97.557	102.425	130.076	136.567
169	449	2432	3234	1730	0.053%	97.340	102.197	129.787	136.263
170	456	1032	2658	1154	0.022%	118.398	124.306	157.864	165.741
171	456	4560	2660	1156	0.097%	118.398	124.306	157.864	165.741
172	457	1336	2664	1160	0.028%	118.139	124.034	157.518	165.378
173	458	1640	2670	1166	0.035%	117.881	123.763	157.174	165.017
174	459	1944	2676	1172	0.041%	117.624	123.493	156.832	164.658
175	466	1888	3356	1852	0.039%	93.789	98.469	125.052	131.292
176	466	4744	3358	1854	0.099%	93.789	98.469	125.052	131.292
177	467	176	3362	1858	0.004%	93.588	98.258	124.784	131.011
178	467	3032	3364	1860	0.063%	93.588	98.258	124.784	131.011
179	468	1320	3370	1866	0.027%	93.388	98.048	124.518	130.731
180	468	4176	3372	1868	0.087%	93.388	98.048	124.518	130.731
181	469	2464	3378	1874	0.051%	93.189	97.839	124.252	130.452
182	470	752	3196	1692	0.016%	98.461	103.374	131.281	137.832
183	470	3776	3198	1694	0.078%	98.461	103.374	131.281	137.832
184	470	752	3384	1880	0.016%	92.991	97.631	123.988	130.175
185	470	3608	3386	1882	0.075%	92.991	97.631	123.988	130.175
186	471	2568	3204	1700	0.053%	98.252	103.155	131.002	137.539
187	471	1896	3392	1888	0.039%	92.793	97.424	123.725	129.898
188	471	4752	3394	1890	0.098%	92.793	97.424	123.725	129.898
189	472	1360	3210	1706	0.028%	98.044	102.936	130.725	137.248
190	472	4384	3212	1708	0.090%	98.044	102.936	130.725	137.248
191	472	184	3398	1894	0.004%	92.597	97.217	123.462	129.623
192	472	3040	3400	1896	0.063%	92.597	97.217	123.462	129.623
193	473	152	3216	1712	0.003%	97.836	102.718	130.449	136.958
194	473	3176	3218	1714	0.065%	97.836	102.718	130.449	136.958
195	474	1968	3224	1720	0.040%	97.630	102.502	130.173	136.669
196	475	760	3230	1726	0.016%	97.424	102.286	129.899	136.381

197	475	3784	3232	1728	0.078%	97.424	102.286	129.899	136.381
198	476	2576	3238	1734	0.053%	97.220	102.071	129.626	136.095
199	478	1840	2660	1156	0.037%	118.327	124.232	157.769	165.642
200	479	2648	2666	1162	0.054%	118.080	123.972	157.440	165.296
201	480	3456	2672	1168	0.070%	117.834	123.714	157.112	164.952
202	481	568	2676	1172	0.012%	117.589	123.457	156.785	164.609
203	481	4264	2678	1174	0.086%	117.589	123.457	156.785	164.609
204	493	184	3352	1848	0.004%	93.867	98.551	125.156	131.402
205	493	3208	3354	1850	0.063%	93.867	98.551	125.156	131.402
206	494	2000	3360	1856	0.039%	93.677	98.352	124.903	131.136
207	494	5024	3362	1858	0.099%	93.677	98.352	124.903	131.136
208	495	792	3366	1862	0.016%	93.488	98.153	124.651	130.871
209	495	3816	3368	1864	0.075%	93.488	98.153	124.651	130.871
210	496	1936	3196	1692	0.038%	98.483	103.397	131.311	137.863
211	496	2608	3374	1870	0.051%	93.300	97.955	124.399	130.607
212	497	1232	3202	1698	0.024%	98.285	103.189	131.046	137.586
213	497	4424	3204	1700	0.087%	98.285	103.189	131.046	137.586
214	497	1400	3380	1876	0.027%	93.112	97.758	124.149	130.344
215	497	4424	3382	1878	0.087%	93.112	97.758	124.149	130.344
216	498	528	3208	1704	0.010%	98.087	102.982	130.783	137.309
217	498	3720	3210	1706	0.073%	98.087	102.982	130.783	137.309
218	498	192	3386	1882	0.004%	92.925	97.562	123.900	130.083
219	498	3216	3388	1884	0.063%	92.925	97.562	123.900	130.083
220	499	1672	2656	1152	0.033%	118.499	124.413	157.999	165.883
221	499	3016	3216	1712	0.059%	97.891	102.776	130.521	137.034
222	499	2008	3394	1890	0.039%	92.739	97.366	123.652	129.822
223	499	5032	3396	1892	0.098%	92.739	97.366	123.652	129.822
224	500	2984	2662	1158	0.058%	118.262	124.164	157.683	165.552
225	500	2312	3222	1718	0.045%	97.695	102.570	130.260	136.760
226	500	800	3400	1896	0.016%	92.553	97.172	123.404	129.562
227	500	3824	3402	1898	0.074%	92.553	97.172	123.404	129.562
228	501	432	2666	1162	0.008%	118.026	123.916	157.369	165.221
229	501	4296	2668	1164	0.083%	118.026	123.916	157.369	165.221
230	501	1608	3228	1724	0.031%	97.500	102.365	130.000	136.487
231	501	4800	3230	1726	0.093%	97.500	102.365	130.000	136.487
232	502	1744	2672	1168	0.034%	117.791	123.669	157.055	164.892
233	502	904	3234	1730	0.018%	97.306	102.161	129.741	136.215
234	502	4096	3236	1732	0.079%	97.306	102.161	129.741	136.215
235	503	3056	2678	1174	0.059%	117.557	123.423	156.743	164.564

需要说明的是，当 $d_{scr} = 336 \times k$ 时， d_{scr} 个加扰比特经 PCS 处理和 OFEC 编码，第一交织处理后，得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特，其经过 DP-16QAM 符号映射和偏振划分刚好得到 172032 个双偏振符号，DSP 成帧后刚好得到一个超帧。也就是，当 $d_{scr} = 336 \times k$ 时，一个超帧对

应第一数据处理每次从数据帧中获取的 r 行的数据,简化了数据处理。当 $d_{scr} \neq 336 \times k$ 时,可通过将 F_{DSP} 个超帧作为一个组合,使得 F_{DSP} 个超帧对应 F_{Frame} 个第一数据处理输出的 $d_{scr} \times F_{Frame}$ 比特,此时有 $d_{scr} \times F_{Frame} = 336 \times k \times F_{DSP}$,其中 F_{Frame} 和 F_{DSP} 为正整数。通过设计每个超帧中所插入的RES保留符号,可用以收端识别 F_{DSP} 个超帧,以实现识别出 F_{Frame} 个第一数据处理输出的 $d_{scr} \times F_{Frame}$ 比特对应的边界。

通过上面对第一数据处理和第二数据处理的介绍可知,对于未来城域电信传输和城域DCI互联场景,将采用FEC编码并结合PCS技术以满足更长传输距离。PCS处理的引入,要求在符号映射操作时,映射到一个调制符号的符号位(sign bits)比特是0、1等概的,且映射到一个调制符号的幅度位(amplitude bits)是0、1不等概的。以FEC编码具体采用OFEC编码为例,为了尽量避免对现有OFEC编码、OFEC交织器的影响,需要在OFEC编码之后、符号映射操作之前引入一个新的交织器,使得经过PCS处理得到的0和1不等概的比特可以映射到调制符号的幅度位比特,实现了在保持星座点位置不变下改变星座点出现的概率,使其非均匀分布,提升了整体性能,以满足未来更长传输距离需求。并且,随着PCS处理的引入,使得进行PCS处理和FEC编码的信息比特长度不再是3552,而是更小的数值,此时需要重新设计从数据帧获取数据的行数,进行CRC校验和插入填充比特的数量,以使得整体数据处理操作更简单,复杂度更低,功耗更低。

下面结合上面对第一数据处理和第二数据处理的介绍几个具体实施例。

实施例1:

考虑800G速率的数据帧,一个数据帧包含512行,每行包含10280比特。考虑表2中的第10组参数组合 $\{r_{Frame} = 110, d_{CP} = 848, k = 3368\}$ 。第一数据处理从数据帧中获取 $r_{Frame} = 110$ 行的数据,总共得到 $d_{in} = 10280 \times r_{Frame} = 1130800$ 个比特。考虑 $p = 22$ 。

图17(a)为本申请实施例中对数据帧进行CRC的一种实施方式示意图。如图17(a)所示,先对每 $\frac{r_{Frame}}{p} = \frac{110}{22} = 5$ 行的 $10280 \times \frac{r_{Frame}}{p} = 51400$ 个比特进行CRC-32校验,加入总共 $d_{CRC} = 22 \times 32 = 704$ 个CRC校验比特;然后插入 $d_{PAD} = 144$ 个填充(Pad)比特,再进行加扰,得到 $d_{scr} = 10280 \times r_{Frame} + d_{CRC} + d_{PAD} = 1131648$ 个加扰比特。CRC校验和插入填充比特对应的冗余为 $OH_{CP} = \frac{d_{scr}}{d_{in}} - 1 = 0.075\%$ 。

如图4所示,第二数据处理从第一数据处理输出的比特数据流中获取多个比特,轮询送入 $L = 4$ 个PCS和OFEC编码进行PCS处理和OFEC编码。如图5所示,PCS和OFEC编码将输入的 $k = 3368$ 个比特中的 $k_{PCS} = k - 1504 = 1864$ 个比特送入PCS处理得到2048个比特。 $k = 3368$ 个比特包含 $k_{PCS} = k - 1504 = 1864$ 个比特和1504个比特。

PCS处理会采用LUT查找表,每次将输入的 $k_{LUT} = 233$ 个比特进行LUT映射输出对应的 $n_{LUT} = 256$ 个比特,连续操作8次上述LUT映射后将总共 $k - 1504 = 1864$ 个比特进行LUT映射输出对应的2048个比特。

PCS处理输出的2048个比特,结合1504个比特,总共 $2048 + 1504 = 3552$ 个比特送入OFEC编码进行编码插入544个OFEC校验比特,得到总共4096个OFEC编码后的比特,即第四比特集合。4096个比特采用如图11(a)所示的方阵集合表示,其中经PCS处理的2048个比特对应如图11(a)所示的8个比特方阵 $B_{0,0}, B_{0,1}, B_{0,2}, B_{0,3}, B_{1,0}, B_{1,1}, B_{1,2}, B_{1,3}$ 。

对第四比特集合(4096个比特)进行第一交织处理,得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的128个比特中第 c_1 个比特来自第四比特集合中第 r_0 行128个比特中第 c_0 个比特,

$c_1 = (c_0\% \theta + (r_1 + 4 \times \Delta)\% \beta)\% \theta + \left\lfloor \frac{c_0}{\theta} \right\rfloor \times \theta$; 其中 $0 \leq c_0 < 128, 0 \leq c_1 < 128, \theta$ 是4的整数倍,且 $m_0 \times 8$

能被 θ 整除, Δ 为整数, β 是4的整数倍,

$$c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{m_0 \times 4} \right\rfloor \right)。$$

这里 $r_0 = r_1$ 。采用第一交织处理,得到交织后的第五比特集合中任一个比特方阵 $T_{i_0 j_0}$ ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的第二比特集合 (2048 个比特) 中的 1 个比特, 其中, $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 1131648$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) =$

- 5 1376256个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍, 有利于硬件实现。OFEC 交织输出的1376256个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号, 并进行 DSP 成帧, 即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS, 得到一个包含 175104 个双偏振符号的超帧, 然后进行发送。
- 10 给定数据帧的速率为 W_{bit} Gbit/s, 对应的波特率为 $W_{baud} = W_{bit} \times (10280 \times r_{Frame} + d_{CP}) / ((10280 \times r_{Frame}) \times (4096/k) \times (57/56) / 8 \text{ Gbaud})$ 。对于 Flex0-8e、Flex0-8 数据帧类型, 其对应波特率分别为 124.7Gbaud 和 130.9Gbaud。

实施例 2:

- 15 考虑 600G 速率的数据帧, 一个数据帧包含 384 行, 每行包含 10280 比特。考虑表 2 中的第 5 组参数组合 $\{r_{Frame} = 87, d_{CP} = 744, k = 2664\}$ 。第一数据处理从数据帧中获取 $r = 87$ 行的数据, 总共得到 $d_{in} = 10280 \times r_{Frame} = 894360$ 个比特。考虑 $p = 3$ 。先对每 $\frac{r_{Frame}}{p} = \frac{87}{3} = 29$ 行的 $10280 \times \frac{r_{Frame}}{p} = 298120$ 个比特进行 CRC-32 校验, 加入总共 $d_{CRC} = 3 \times 32 = 96$ 个 CRC 校验比特; 然后插入 $d_{PAD} = 648$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{scr} = 10280 \times r + d_{CRC} + d_{PAD} = 895104$ 个加扰比特。CRC 校验
- 20 和插入填充比特对应的冗余为 $OH_{CP} = \frac{d_{scr}}{d_{in}} - 1 = 0.083\%$ 。

如图 4 所示, 第二数据处理从第一数据处理输出的比特数据流中获取多个比特, 轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示, PCS 和 OFEC 编码将输入的 $k = 2664$ 个比特中的 $k_{PCS} = k - 1504 = 1160$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 2664$ 个比特包含 $k_{PCS} = k - 1504 = 1160$ 个比特和 1504 个比特。

- 25 PCS 处理会采用 LUT 查找表, 每次将输入的 $k_{LUT} = 145$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 256$ 个比特, 连续操作 8 次上述 LUT 映射后将总共 $k - 1504 = 1160$ 个比特进行 LUT 映射输出对应的2048个比特。

PCS 处理输出的2048个比特结合 1504 个比特总共 $2048+1504=3552$ 个比特送入 OFEC 编码进行编码插入 544 个 OFEC 校验比特, 得到总共 4096 个 OFEC 编码后的比特, 即第四比特集合。4096 个比特采用如图 11 (a) 所示的方阵集合表示, 其中经 PCS 处理的 2048 个比特对应如图 11 (a) 所示的 8 个比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 。

对第四比特集合进行第一交织处理得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自第四比特集合中第 r_0 行 128 个比特中第 c_0 个比特, $c_1 = (c \% \theta + (r_1 + 2 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$; 其中 $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为

- 35 整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(\left\lfloor \frac{c_0}{32} \right\rfloor \% 2 \right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64。$$

这里 $r_0 = r_1$ 。采用第一交织处理, 得到第五比特集合中任一个方阵 $T_{i_0 j_0}$ ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的第二比特集合 (2048 个比

特) 中的 1 个比特, 其中, $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 895104$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍, 有利于硬件实现。

5 OFEC 交织输出的 1376256 个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号, 并进行 DSP 成帧, 即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS, 得到一个包含 175104 个双偏振符号的超帧, 然后进行发送。

给定数据帧的速率为 W_{bit} Gbit/s, 对应的波特率为 $W_{baud} = W_{bit} \times (10280 \times r_{Frame} + d_{CP}) / ((10280 \times r_{Frame}) \times (4096/k) \times (57/56)) / 8$ Gbaud。对于 Flex0-6e、Flex0-6 数据帧类型, 其对应波特率分别为 118.2 Gbaud 和 124.1 Gbaud。

10

实施例 3:

考虑 600G 速率的数据帧, 一个数据帧包含 384 行, 每行包含 10280 比特。考虑表 2 中的第 6 组参数组合 $\{r_{Frame} = 105, d_{CP} = 504, k = 3214\}$ 。第一数据处理从数据帧中获取 $r_{Frame} = 105$ 行的数据,

15 总共得到 $d_{in} = 10280 \times r_{Frame} = 1079400$ 个比特。考虑 $p = 15$ 。先对每 $\frac{r_{Frame}}{p} = \frac{105}{15} = 7$ 行的 $10280 \times \frac{r_{Frame}}{p} = 71960$ 个比特进行 CRC-32 校验, 加入总共 $d_{CRC} = 15 \times 32 = 480$ 个 CRC 校验比特; 然后插入 $d_{PAD} = 24$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{scr} = 10280 \times r_{Frame} + d_{CRC} + d_{PAD} = 1079904$ 比特。CRC 校验和插入填充比特对应的冗余为 $OH_{CP} = \frac{d_{scr}}{d_{in}} - 1 = 0.047\%$ 。

如图 4 所示, 第二数据处理从第一数据处理输出的比特数据流中获取多个比特, 轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示, PCS 和 OFEC 编码将输入的 $k = 3214$ 个比特中的 $k_{PCS} = k - 1504 = 1710$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 3214$ 个比特包含 $k_{PCS} = k - 1504 = 1710$ 个比特和 1504 个比特。

20

PCS 处理会采用 LUT 查找表, 每次将输入的 $k_{LUT} = 855$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 1024$ 个比特, 连续操作 2 次上述 LUT 映射后将总共 $k - 1504 = 1710$ 个比特进行 LUT 映射输出对应的 2048 个比特。

25

PCS 处理输出的 2048 个比特结合 1504 个比特总共 $2048 + 1504 = 3552$ 个比特送入 OFEC 编码进行编码插入 544 个 OFEC 校验比特, 得到总共 4096 个 OFEC 编码后的比特, 即第四比特集合。4096 个比特采用如图 11 (b) 所示的方阵集合表示, 其中经 PCS 处理的 2048 个比特对应如图 11 (b) 所示的 8 个比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 。

30

对第四比特集合进行第一交织处理得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自第四比特集合中第 r_0 行 128 个比特中第 c_0 个比特, $c_1 = (c_0 \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c_0}{\theta} \right\rfloor \times \theta$; 其中 $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64。$$

35 这里 $r_0 = r_1$ 。采用第一交织处理得到交织后的方阵集合中任一个方阵 T_{i_0, j_0} ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的 2048 个比特中的 1 个比特, 其中 $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 1079904$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍，有利于硬件实现。OFEC 交织输出的1376256个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号，并进行 DSP 成帧，即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS，得到一个包含 175104 个双偏振符号的超帧，然后进行发送。

给定数据帧的速率为 W_{bit} Gbit/s，对应的波特率为 $W_{baud} = W_{bit} \times (10280 \times r_{Frame} + d_{CP}) / (10280 \times r_{Frame}) \times (4096/k) \times (57/56) / 8$ Gbaud。对于 Flex0-6e 数据帧类型，其对应波特率为 130.6Gbaud。

10

实施例 4:

考虑 600G 速率的数据帧，一个数据帧包含 384 行，每行包含 10280 比特。考虑表 3 中的第 2 组参数组合 $\{r_{Frame} = 261, d_{CP} = 2232, k = 2664\}$ 。第一数据处理从数据帧中获取 $r_{Frame} = 261$ 行的数据，总共得到 $d_{in} = 10280 \times r_{Frame} = 2683080$ 个比特。考虑 $p = 29$ 。先对每 $\frac{r_{Frame}}{p} = \frac{261}{29} = 9$ 行的 $10280 \times$

15 $\frac{r_{Frame}}{p} = 92520$ 个比特进行 CRC-32 校验，加入总共 $d_{CRC} = 29 \times 32 = 928$ 个 CRC 校验比特；然后插入 $d_{PAD} = 1304$ 个填充 (Pad) 比特，再进行加扰，得到 $d_{scr} = 10280 \times r_{Frame} + d_{CRC} + d_{PAD} = 2685312$ 比特。CRC 校验和插入填充比特对应的冗余为 $OH_{CP} = \frac{d_{scr}}{d_{in}} - 1 = 0.083\%$ 。

如图 4 所示，第二数据处理从第一数据处理输出的比特数据流中获取多个比特，轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示，PCS 和 OFEC 编码将输入的 $k = 2664$ 个比特中的 $k_{PCS} = k - 1504 = 1160$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 2664$ 个比特包含 $k_{PCS} = k - 1504 = 1160$ 个比特和 1504 个比特。

PCS 处理会采用 LUT 查找表，每次将输入的 $k_{LUT} = 290$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 512$ 个比特，连续操作 4 次上述 LUT 映射后将总共 $k - 1504 = 1160$ 个比特进行 LUT 映射输出对应的2048个比特。需要说明的是，在另一些具体实施例中，所述 PCS 处理采用 LUT 查找表每次将输入的 $k_{LUT} = 145$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 256$ 个比特，连续操作 8 次上述 LUT 映射后将总共 $k - 1504 = 1160$ 个比特进行 LUT 映射输出对应的2048个比特。

PCS 处理输出的2048个比特结合 1504 个比特总共 $2048+1504=3552$ 个比特送入 OFEC 编码进行编码插入 544 个 OFEC 校验比特，得到总共 4096 个 OFEC 编码后的比特，即第四比特集合。4096 个比特采用如图 11 (b) 所示的方阵集合表示，其中经 PCS 处理的 2048 个比特对应如图 11 (b) 所示的 8 个比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,4}$ 、 $B_{0,5}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,4}$ 、 $B_{1,5}$ 。

对第四比特集合进行第一交织处理得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自第四比特集合中第 r_0 行 128 个比特中第 c_0 个比特， $c_1 = (c_0 \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c_0}{\theta} \right\rfloor \times \theta$ ；其中 $0 \leq c_0 < 128$ ， $0 \leq c_1 < 128$ ， θ 是 4 的整数倍，且 $m_0 \times 8$ 能被 θ 整除， Δ 为整数， β 是 4 的整数倍，

$$35 \quad c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64。$$

这里 $r_0 = r_1$ 。采用第一交织处理得到交织后的方阵集合中任一个方阵 T_{i_0, j_0} ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的 2048 个比特中的 1 个比特，其中 $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 2685312$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 4128768$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍，有利于硬件实现。OFEC 交织输出的4128768个比特送入 DP-16QAM 符号映射和偏振划分得到 516096=172032×3个双偏振符号。所述 516096=172032×3个双偏振符号中每 172032 个双偏振符号进行 DSP 成帧，即对 X 偏振和 Y 偏振符号插入 FAW 帧对齐符号、TS 训练符号、RES 保留符号和 PS 导频符号，得到一个包含 175104 个双偏振符号的超帧，总共得到 3 个超帧，然后进行发送。典型的，在所述一个超帧中包含 74 个 RES 保留符号。

需要说明的是，所述 $d_{scr} \neq 336 \times k$ ，考虑 $d_{scr} \times F_{Frame} = 336 \times k \times F_{DSP}$ ，其中 $F_{Frame} = 1$ 且 $F_{DSP} = 3$ ，可通过将上述 $F_{DSP} = 3$ 个超帧作为一个组合，使得 F_{DSP} 个超帧对应 $F_{Frame} = 1$ 个第一数据处理输出的 $d_{scr} \times F_{Frame} = 4128768$ 比特。通过设计每个超帧中所插入的 RES 保留符号，可用以收端识别 $F_{DSP} = 3$ 个超帧，以实现识别出 $F_{Frame} = 1$ 个第一数据处理输出的 $d_{scr} \times F_{Frame} = 4128768$ 比特对应的边界。

需要说明的是，所述“设计每个超帧中所插入的 RES 保留符号”的一种具体实施方式如下：对于所述 $F_{DSP} = 3$ 个超帧，第 0 个超帧中连续 22 个 RES 保留符号为特定的第一标识序列（也可称为第一标识符号），第 1 个和第 2 个超帧中连续 22 个 RES 保留符号为特定的第二标识序列（也可称为第二标识符号），所述第一标识序列和所述第二标识序列不一样；接收端在进行 DSP 超帧对齐后，可通过识别所述第一标识序列和所述第二标识序列获取所述 $F_{DSP} = 3$ 个超帧一个组合的边界。所述第一标识、第二标识也可称为所述 F_{DSP} 超帧对齐标识（ F_{DSP} superframe alignment signal）。此时，除去所述第一标识或第二标识剩余的 RES 保留符号也称为固定填充（Fixed Stuff, FS）。在一些具体实现中，所述第一标识序列、第二标识序列具体取值如下表 4 所示，其对应的两个偏振方向上的 16QAM 符号可将所述序列进行 16QAM 符号映射获得。

表 4

序号	第一标识序列		第二标识序列	
	X 偏振	Y 偏振	X 偏振	Y 偏振
0	1000	0110	0001	0000
1	1101	1110	1010	1111
2	1011	1100	0110	1100
3	1000	1101	1100	0100
4	1111	1110	0100	0111
5	0001	1110	0000	1111
6	1100	1101	1011	0110
7	0010	1011	1110	1101
8	0100	0010	0011	1101
9	0000	1100	0111	0010
10	1010	0000	1001	0000
11	1001	1000	1001	1100

12	0011	0001	1110	1010
13	0101	0111	1000	1000
14	0111	0000	1111	1001
15	0010	0110	0001	0011
16	0011	1110	1110	0100
17	0100	1011	1010	0110
18	1111	0101	0100	0010
19	1111	1110	0101	0001
20	1000	0001	1011	0110
21	0101	1110	1101	0111

给定数据帧的速率为 W_{bit} Gbit/s，对应的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times (10280 \times r_{\text{Frame}} + d_{\text{CP}}) / (10280 \times r_{\text{Frame}}) \times (4096/k) \times (57/56)/8$ Gbaud。对于 Flex0-6e 数据帧类型，其对应波特率为 118.203Gbaud。对于 Flex0-6 数据帧类型，其对应波特率为 124.102Gbaud。

需要说明的是，在一些具体场景，所述包含 172032 个双偏振符号的超帧另称为多帧(multi-frame)，
5 所述包含 172032×3 个双偏振符号的 $F_{\text{DSP}} = 3$ 个多帧另称为 1 个超帧。

实施例 5:

考虑 800G 速率的数据帧，一个数据帧包含 512 行，每行包含 10280 比特。考虑表 3 中的第 3 组参数组合 $\{r_{\text{Frame}} = 261, d_{\text{CP}} = 1560, k = 3196\}$ 。第一数据处理从数据帧中获取 $r_{\text{Frame}} = 261$ 行的数据，
10 总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 2683080$ 个比特。

考虑 $p = 29$ 。先对每 $\frac{r_{\text{Frame}}}{p} = \frac{261}{29} = 9$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 92520$ 个比特进行 CRC-32 校验，加入总共 $d_{\text{CRC}} = 29 \times 32 = 928$ 个 CRC 校验比特；然后插入 $d_{\text{PAD}} = 632$ 个填充 (Pad) 比特，再进行加扰，得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 2684640$ 比特。CRC 校验和插入填充比特对应的冗余为 $\text{OH}_{\text{CP}} = \frac{d_{\text{scr}}}{d_{\text{in}}} - 1 = 0.058\%$ 。

15 如图 4 所示，第二数据处理从第一数据处理输出的比特数据流中获取多个比特，轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示，PCS 和 OFEC 编码将输入的 $k = 3196$ 个比特中的 $k_{\text{PCS}} = k - 1504 = 1692$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 3196$ 个比特包含 $k_{\text{PCS}} = k - 1504 = 1692$ 个比特和 1504 个比特。

20 PCS 处理会采用 LUT 查找表，每次将输入的 $k_{\text{LUT}} = 423$ 个比特进行 LUT 映射输出对应的 $n_{\text{LUT}} = 512$ 个比特，连续操作 4 次上述 LUT 映射后将总共 $k - 1504 = 1692$ 个比特进行 LUT 映射输出对应的 2048 个比特。

25 PCS 处理输出的 2048 个比特，结合 1504 个比特，总共 $2048 + 1504 = 3552$ 个比特送入 OFEC 编码进行编码插入 544 个 OFEC 校验比特，得到总共 4096 个 OFEC 编码后的比特，即第四比特集合。4096 个比特采用如图 11 (a) 所示的方阵集合表示，其中经 PCS 处理的 2048 个比特对应如图 11 (a) 所示的 8 个比特方阵 $B_{0,0}, B_{0,1}, B_{0,2}, B_{0,3}, B_{1,0}, B_{1,1}, B_{1,2}, B_{1,3}$ 。

对第四比特集合 (4096 个比特) 进行第一交织处理，得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自第四比特集合中第 r_0 行 128 个比特中第 c_0 个比特，

$c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$; 其中 $0 \leq c_0 < 128$, $0 \leq c_1 < 128$, θ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_0 \% (m_0 \times 4)}{2} \right\rfloor \times 4 + c_0 \% 2 + 2 \times \left(1 - \left\lfloor \frac{c_0}{(m_0 \times 4)} \right\rfloor \right)。$$

这里 $r_0 = r_1$ 。采用第一交织处理, 得到交织后的第五比特集合中任一个比特方阵 $T_{i_0 j_0}$ ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的第二比特集合 (2048 个比特) 中的 1 个比特, 其中, $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 2684640$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 3440640$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍, 有利于硬件实现。

OFEC 交织输出的 3440640 个比特送入 DP-16QAM 符号映射和偏振划分得到 430080 个双偏振符号。

需要说明的是, 所述 $d_{scr} \neq 336 \times k$, 考虑 $d_{scr} \times F_{Frame} = 336 \times k \times F_{DSP}$, 其中 $F_{Frame} = 2$ 且 $F_{DSP} = 5$ 。也就是, $F_{Frame} = 2$ 个第一数据处理输出的 $d_{scr} \times F_{Frame} = 5369280$ 比特, 对应 860160 个双偏振符号。所述 860160 = 172032 × 5 个双偏振符号中每 172032 个双偏振符号进行 DSP 成帧, 即对 X 偏振和 Y 偏振符号插入 FAW 帧对齐符号、TS 训练符号、RES 保留符号和 PS 导频符号, 得到一个包含 175104 个双偏振符号的超帧, 总共得到 5 个超帧, 然后进行发送。典型的, 在所述一个超帧中包含 74 个 RES 保留符号。

上述 $F_{DSP} = 5$ 个超帧作为一个组合, 使得 $F_{DSP} = 5$ 个超帧对应 $F_{Frame} = 2$ 个第一数据处理输出的 $d_{scr} \times F_{Frame} = 5369280$ 比特。通过设计每个超帧中所插入的 RES 保留符号, 可用以收端识别 $F_{DSP} = 5$ 个超帧, 以实现识别出 $F_{Frame} = 2$ 个第一数据处理输出的 $d_{scr} \times F_{Frame} = 5369280$ 比特对应的边界。

需要说明的是, 所述“设计每个超帧中所插入的 RES 保留符号”的一种具体实施方式如下: 对于所述 $F_{DSP} = 5$ 个超帧, 第 0 个超帧中连续 22 个 RES 保留符号为特定的第一标识序列, 第 1 个至第 4 个超帧中连续 22 个 RES 保留符号为特定的第二标识序列, 所述第一标识序列和所述第二标识序列不一样; 接收端在进行 DSP 超帧对齐后, 可通过识别所述第一标识序列和所述第二标识序列获取所述 $F_{DSP} = 5$ 个超帧一个组合的边界。所述“设计每个超帧中所插入的 RES 保留符号”的另一种具体实施方式如下: 对于所述 $F_{DSP} = 5$ 个超帧, 第 0 个和第 1 个超帧中连续 22 个 RES 保留符号为特定的第一标识序列 (也可称为第一标识符号), 第 2 个至第 4 个超帧中连续 22 个 RES 保留符号为特定的第二标识序列 (也可称为第二标识符号), 所述第一标识序列和所述第二标识序列不一样; 接收端在进行 DSP 超帧对齐后, 可通过识别所述第一标识序列和所述第二标识序列获取所述 $F_{DSP} = 5$ 个超帧一个组合的边界。所述第一标识、第二标识也可称为所述 F_{DSP} 超帧对齐标识 (F_{DSP} superframe alignment signal, SFAS)。此时, 除去所述第一标识或第二标识剩余的 RES 保留符号也称为固定填充 (Fixed Stuff, FS)。在一些具体实现中, 所述第一标识符号、第二标识符号具体取值如表 5 所示。

表 5

序号	第一标识符号		第二标识符号	
	X 偏振	Y 偏振	X 偏振	Y 偏振
0	3-3j	-1+3j	-3-1j	-3-3j
1	1-1j	1+3j	3+3j	1+1j
2	3+1j	1-3j	-1+3j	1-3j
3	3-3j	1-1j	1-3j	-1-3j
4	1+1j	1+3j	-1-3j	-1+1j

5	$-3-1j$	$1+3j$	$-3-3j$	$1+1j$
6	$1-3j$	$1-1j$	$3+1j$	$-1+3j$
7	$-3+3j$	$3+1j$	$1+3j$	$1-1j$
8	$-1-3j$	$-3+3j$	$-3+1j$	$1-1j$
9	$-3-3j$	$1-3j$	$-1+1j$	$-3+3j$
10	$3+3j$	$-3-3j$	$3-1j$	$-3-3j$
11	$3-1j$	$3-3j$	$3-1j$	$1-3j$
12	$-3+1j$	$-3-1j$	$1+3j$	$3+3j$
13	$-1-1j$	$-1+1j$	$3-3j$	$3-3j$
14	$-1+1j$	$-3-3j$	$1+1j$	$3-1j$
15	$-3+3j$	$-1+3j$	$-3-1j$	$-3+1j$
16	$-3+1j$	$1+3j$	$1+3j$	$-1-3j$
17	$-1-3j$	$3+1j$	$3+3j$	$-1+3j$
18	$1+1j$	$-1-1j$	$-1-3j$	$-3+3j$
19	$1+1j$	$1+3j$	$-1-1j$	$-3-1j$
20	$3-3j$	$-3-1j$	$3+1j$	$-1+3j$
21	$-1-1j$	$1+3j$	$1-1j$	$-1+1j$

给定数据帧的速率为 W_{bit} Gbit/s, 对应的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times (10280 \times r_{\text{Frame}} + d_{\text{CP}}) / (10280 \times r_{\text{Frame}}) \times (4096/k) \times (57/56)/8$ Gbaud。对于 Flex0-8e 数据帧类型, 其对应波特率为 131.337Gbaud。

需要说明的是, 在一些具体场景, 所述包含 175104 个双偏振符号的超帧另称为多帧(multi-frame), 所述包含 175104×5 个双偏振符号的 $F_{\text{DSP}} = 5$ 个超帧另称为 1 个超帧。

实施例 6:

考虑 600G 速率的数据帧, 一个数据帧每行包含 q 比特。第一数据处理从数据帧中获取 r_{Frame} 行的数据, 总共得到 $d_{\text{in}} = q \times r_{\text{Frame}}$ 个比特。之后, 对第一数据进行 CRC 和/或插入填充比特得到第二数据。其中, 在 CRC 操作中会加入 d_{CRC} 个 CRC 校验比特, 在插入填充操作中会插入 d_{PAD} 个填充(Pad)比特。进而, 对第二数据进行加扰得到第三数据, 第三数据的比特数量 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CP}}$ 个加扰比特, 其中 $d_{\text{CP}} = d_{\text{CRC}} + d_{\text{PAD}}$ 。加扰后的比特数据进行包含 PCS 和 OFEC 编码、第一交织处理、OFEC 交织、16QAM 符号映射和偏振划分得到一个包含 172032 个双偏振符号, 并进行冗余为 $\frac{1}{56}$ 的 DSP 成帧得到一个包含 175104 个双偏振符号的超帧, 有如下公式 1:

$$d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CP}} = 336 \times k$$

所述数据处理对应的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times (q \times r_{\text{Frame}} + d_{\text{CP}}) / (q \times r_{\text{Frame}}) \times (4096/k) \times (57/56)/8$ Gbaud, 其中 W_{bit} 为所述 600G 数据帧对应的具体比特速率值。本实施例中, 采用 PCS 的所述 600G 速率场景下, 其对应的波特率 W_{baud} 跟现有 800G ZR 的波特率场景一样, 有如下公式 2:

$$3 \times \frac{q \times r_{\text{Frame}} + d_{\text{CP}}}{q \times r_{\text{Frame}}} \times \left(\frac{4096}{k} \right) = 4 \times \frac{116 \times 10280 + 992}{116 \times 10280} \times \left(\frac{4096}{3552} \right)$$

结合公式 1 和公式 2, 有

$$3 \times \frac{336 \times k}{q \times r_{\text{Frame}}} \times \left(\frac{4096}{k} \right) = 4 \times \frac{116 \times 10280 + 992}{116 \times 10280} \times \left(\frac{4096}{3552} \right)$$

即有如下公式 3:

$$q \times r_{\text{Frame}} = \frac{3 \times 336 \times 116 \times 10280 \times 3552}{4 \times (116 \times 10280 + 992)} = 894360$$

考虑所述 600G 速率的一个数据帧包含 512 行, 每行包含 $q = 10280$ 比特。此时, 有:

$$r_{\text{Frame}} = \frac{894360}{q} = 87$$

需要说明的是, 本实施例中第一数据处理从数据帧中获取 $d_{\text{in}} = q \times r_{\text{Frame}} = 894360$ 个比特的第一

- 5 数据, 第一数据包括数据帧中的 $r_{\text{Frame}} = \frac{894360}{q}$ 行, 每行包含 q 个比特。本实施例中以每行包含 $q = 10280$ 个比特为例。所采用的 q 取值也可以为 257 整数倍的其他整数, 如 $q = 2056$ 。当 $q = 2056$ 时, $r_{\text{Frame}} = \frac{894360}{q} = 435$ 。

考虑 CRC 校验和插入填充比特对应的冗余为 $\text{OH}_{\text{CP}} = \frac{d_{\text{scr}}}{d_{\text{in}}} - 1 > 0$, 有 $q \times r_{\text{Frame}} + d_{\text{CP}} > q \times r_{\text{Frame}}$ 。

- 10 再结合公式 1 和公式 3, 有 $336 \times k > 894360$, 即整数 $k \geq 2662$ 。

- PCS 处理会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。对于 PCS 处理的输出比特数量 n_{LUT} 不能太小, 否则会导致整体 PCS 处理导致性能损失。PCS 处理的输出比特数量 n_{LUT} 也不能太大, 否则不利于硬件实现。典型的是, $n_{\text{LUT}} = 128$ 或 256。当 $n_{\text{LUT}} = 128$ 时, PCS 处理连续操作 $2048/128=16$ 次上述 LUT 映射后将总共 $k - 1504$ 个比特 (第一比特集合) 进行 LUT 映射输出对应的 2048 个比特 (对应第二比特集合), 有整数 $k - 1504$ 是 16 的整数倍。当 $n_{\text{LUT}} = 256$ 时, PCS 处理连续操作 $2048/256=8$ 次上述 LUT 映射后将总共 $k - 1504$ 个比特 (第一比特集合) 进行 LUT 映射输出对应的 2048 个比特 (对应第二比特集合), 有整数 $k - 1504$ 是 8 的整数倍。因此, 有整数 $k - 1504$ 是 8 的整数倍。

- 20 由于冗余为 $\text{OH}_{\text{CP}} = \frac{d_{\text{scr}}}{d_{\text{in}}} - 1 = \frac{336 \times k}{894360} - 1$, 整数 k 越大, 冗余 OH_{CP} 越大。典型的, 冗余 OH_{CP} 取值较小, 以提升系统有效性。再结合, 整数 $k \geq 2662$ 且整数 $k - 1504$ 是 8 的整数倍, 整数 k 的典型取值可为 2664 或 2672。其对应的典型参数组合如表 6 所示。

- 25 需要说明的是, 表 6 中序号 1 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 744, k = 2664\}$ 下, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 256, 且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.083\%$ 。表 6 中序号 2 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 3432, k = 2672\}$ 下, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128, 且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.384\%$ 。两种参数组合下, 序号 1 对应的 PCS 查找表具体实现复杂度较高, 但 CRC 和 PAD 冗余 OH_{CP} 较小。序号 2 对应的 PCS 查找表具体实现复杂度较低, 但 CRC 和 PAD 冗余 OH_{CP} 较高。可根据具体应用场景选取采用序号 1 或序号 2 的参数组合进行实现。

表 6

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	
						FlexO-6e	FlexO-6
1	87	744	2664	1160	0.083%	118.203	124.102
2	87	3432	2672	1168	0.384%	118.203	124.102

- 30 需要说明的是, 对于表 6 中序号 2 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 3432, k = 2672\}$ 下, 选取参数 $p=29$ 。对应的, 第一数据处理从数据帧中获取 $r_{\text{Frame}} = 87$ 行的数据, 总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} =$

894360个比特。先对每 $\frac{r_{\text{Frame}}}{p} = \frac{87}{29} = 3$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 30840$ 个比特进行 CRC-32 校验, 加入总共 $d_{\text{CRC}} = 29 \times 32 = 928$ 个 CRC 校验比特; 然后插入 $d_{\text{PAD}} = 2504$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 897792$ 个加扰比特。

如图 4 所示, 第二数据处理从第一数据处理输出的比特数据流中获取多个比特, 轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示, PCS 和 OFEC 编码将输入的 $k = 2672$ 个比特中的 $k_{\text{PCS}} = k - 1504 = 1168$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 2672$ 个比特包含 $k_{\text{PCS}} = k - 1504 = 1168$ 个送入 PCS 处理的比特和剩余的 1504 个比特。

PCS 处理会采用 LUT 查找表, 每次将输入的 $k_{\text{LUT}} = 73$ 个比特进行 LUT 映射输出对应的 $n_{\text{LUT}} = 128$ 个比特, 连续操作 16 次上述 LUT 映射后将总共 $k - 1504 = 1168$ 个比特进行 LUT 映射输出对应的 2048 个比特。

PCS 处理输出的 2048 个比特, 结合 1504 个比特, 总共 $2048 + 1504 = 3552$ 个比特送入 OFEC 编码进行编码插入 544 个 OFEC 校验比特, 得到总共 4096 个 OFEC 编码后的比特, 即第四比特集合。4096 个比特采用如图 11 (b) 所示的方阵集合表示, 其中经 PCS 处理的 2048 个比特对应如图 11 (b) 所示的 8 个比特方阵 $B_{0,0}, B_{0,1}, B_{0,4}, B_{0,5}, B_{1,0}, B_{1,1}, B_{1,4}, B_{1,5}$ 。

所述来自第二比特集合的 2048 个比特在所述 FEC 编码的码字中位置是不连续的。此时, 所述第二比特集合和所述第三比特集合在送入所述 FEC 编码时还经过了编码前交织处理以打乱顺序, 使其编码前的 3552 个比特的排放顺序如下: 先是来自第二比特集合的连续 1024 个比特, 接着是来自第三比特集合的连续 1024 个比特, 再接着是来自第二比特集合的连续 1024 个比特, 最后是来自第三比特集合的连续 480 个比特。

对第四比特集合进行第一交织处理得到交织后的第五比特集合。交织后的第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自第四比特集合中第 r_0 行 128 个比特中第 c_0 个比特, $c_1 = (c_0\theta + (r_1 + 4 \times \Delta)\beta)\theta + \left\lfloor \frac{c_0}{\theta} \right\rfloor \times \theta$; 其中 $0 \leq c_0 < 128, 0 \leq c_1 < 128, \theta$ 是 4 的整数倍, 且 $m_0 \times 8$ 能被 θ 整除, Δ 为整数, β 是 4 的整数倍,

$$c = \left\lfloor \frac{c_0\%32}{2} \right\rfloor \times 4 + c_0\%2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor\%2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64。$$

这里 $r_0 = r_1$ 。采用第一交织处理得到交织后的方阵集合中任一个方阵 T_{i_0,j_0} ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2\%2 + i_2 + 2\right)\%16$ 列的比特是来自经过 PCS 处理得到的 2048 个比特中的 1 个比特, 其中 $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{\text{scr}} = 897792$ 个加扰比特经 PCS 处理和 OFEC 编码以及第一交织处理得到 $\frac{d_{\text{scr}}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍, 有利于硬件实现。OFEC 交织输出的 1376256 个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号, 并进行 DSP 成帧, 即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS, 得到一个包含 175104 个双偏振符号的超帧, 然后进行发送。

给定数据帧的速率为 W_{bit} Gbit/s, 对应的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times (10280 \times r_{\text{Frame}} + d_{\text{CP}}) / (10280 \times r_{\text{Frame}}) \times (4096/k) \times (57/56) / 8$ Gbaud。对于 Flex0-6e 数据帧类型, 其对应波特率为 118.203 Gbaud, 与 800ZR 下的波特率一样。

实施例 7:

图 17 (b) 为本申请实施例中对数据帧进行 CRC 的另一种实施方式示意图。在实施例 6 基础上, 考虑表 6 中序号 2 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 3432, k = 2672\}$, 并选取参数 $p=18$, 参照图 17 (b)

进行理解。对应的,第一数据处理从数据帧中获取 $r_{\text{Frame}} = 87$ 行的数据,总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 894360$ 个比特。考虑整数 $r_{\text{F0}} = \lfloor r_{\text{Frame}}/p \rfloor + 1 = 5$,整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 87 - 5 \times 17 = 2$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 87$ 行数据,对其中最先的 $r_{\text{F0}} \times (p - 1) = 85$ 行中的每 $r_{\text{F0}} = 5$ 行共 $10280 \times r_{\text{F0}} = 51400$ 个比特进行 CRC-32 添加长度 32 比特的校验比特得到 $32 \times (p - 1) = 544$ 个比特的 CRC 校验比特,对最后的 $r_{\text{F1}} = 2$ 行共 $10280 \times r_{\text{F1}} = 20560$ 个比特进行 CRC-32 添加长度 32 比特的校验比特,总共得到 $d_{\text{CRC}} = 32 \times p = 576$ 个比特的 CRC 校验比特。然后插入 $d_{\text{PAD}} = 2856$ 个填充 (Pad) 比特,再进行加扰,得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 897792$ 个加扰比特。

实施例 8:

图 17 (c) 为本申请实施例中对数据帧进行 CRC 的另一种实施方式示意图。在实施例 6 基础上,考虑表 6 中序号 2 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 3432, k = 2672\}$,并选取参数 $p=22$,参照图 17 (c) 进行理解。对应的,第一数据处理从数据帧中获取 $r_{\text{Frame}} = 87$ 行的数据,总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 894360$ 个比特。考虑整数 $r_{\text{F0}} = \lfloor r_{\text{Frame}}/p \rfloor + 1 = 4$,整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 87 - 4 \times 21 = 3$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 87$ 行数据,对其中最先的 $r_{\text{F0}} \times (p - 1) = 84$ 行中的每 $r_{\text{F0}} = 4$ 行共 $10280 \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度 32 比特的校验比特得到 $32 \times (p - 1) = 672$ 个比特的 CRC 校验比特,对最后的 $r_{\text{F1}} = 3$ 行共 $10280 \times r_{\text{F1}} = 30840$ 个比特进行 CRC-32 添加长度 32 比特的校验比特,总共得到 $d_{\text{CRC}} = 32 \times p = 704$ 个比特的 CRC 校验比特。然后插入 $d_{\text{PAD}} = 2728$ 个填充 (Pad) 比特,再进行加扰,得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 897792$ 个加扰比特。需要说明的是,所述对 $r_{\text{Frame}} = 87$ 行数据中最前面 $r_{\text{F0}} \times (p - 1) = 84$ 行中的每 $r_{\text{F0}} = 4$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32。所述 CRC-32 编码、检测操作可直接采用现有 800GZR 中的 CRC-32 操作,其中现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。

需要说明的是,上述场景中考虑的是数据帧每行包含 $q = 10280$ 个比特。在另一些可能的场景中,数据帧每行包含 $q = 2056$ 个比特。作为一个示例,考虑参数组合 $\{q = 2056, r_{\text{Frame}} = 435, d_{\text{CP}} = 3432, k = 2672\}$,并选取参数 $p=22$ 。此时,第一数据处理从数据帧中获取 $r_{\text{Frame}} = 435$ 行的数据,总共得到 $d_{\text{in}} = q \times r_{\text{Frame}} = 894360$ 个比特。考虑整数 $r_{\text{F0}} = \lfloor r_{\text{Frame}}/p \rfloor + 1 = 20$,整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 435 - 20 \times 21 = 15$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 435$ 行数据,对其中在先的 $r_{\text{F0}} \times (p - 1) = 420$ 行中每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特得到 $32 \times (p - 1) = 672$ 个比特的 CRC 校验比特,再对其中最后的 $r_{\text{F1}} = 15$ 行共 $q \times r_{\text{F1}} = 30840$ 个比特进行 CRC-32 添加长度 32 比特的校验比特,总共得到 $d_{\text{CRC}} = 32 \times p = 704$ 个比特的 CRC 校验比特。然后,插入 $d_{\text{PAD}} = 2728$ 个填充 (Pad) 比特,再进行加扰得到 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 897792$ 个加扰比特。

需要说明的是,对于表 6 中序号 1 的参数组合 $\{r_{\text{Frame}} = 87, d_{\text{CP}} = 744, k = 2664\}$,其应用的是数据帧每行包含 $q=10280$ 个比特的场景。对于应用于数据帧每行包含 $q=2056$ 个比特的场景,考虑参数组合 $\{q=2056, r_{\text{Frame}} = 435, d_{\text{CP}} = 744, k = 2664\}$,并选取参数 $p=22$ 。此时,第一数据处理从数据帧中获取 $r_{\text{Frame}} = 435$ 行的数据,总共得到 $d_{\text{in}} = q \times r_{\text{Frame}} = 894360$ 个比特。考虑整数 $r_{\text{F0}} = \lfloor r_{\text{Frame}}/p \rfloor + 1 = 20$,整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 435 - 20 \times 21 = 15$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 435$ 行数据,对其中在先的 $r_{\text{F0}} \times (p - 1) = 420$ 行中的每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特得到 $32 \times (p - 1) = 672$ 个比特的 CRC 校验比特,再对其中最后的 $r_{\text{F1}} = 15$ 行共 $q \times r_{\text{F1}} = 30840$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特,总共得到 $d_{\text{CRC}} = 32 \times p = 704$ 个比特的 CRC 校验比特。然后,插入 $d_{\text{PAD}} = 40$ 个填充 (Pad) 比特,再进行加扰得到 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 895104$ 个加扰比特。

需要说明的是,在对 $r_{\text{Frame}} = 435$ 行数据中在先的 $r_{\text{F0}} \times (p - 1) = 420$ 行中的每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 操作的场景中,该 CRC-32 编码和检测操作可直接采用现有 800GZR 中的 CRC-32 操作,区别在于现有 800GZR 中是对 4 行 10280 列的共 $4 \times 10280 = 41120$ 个比特进行 CRC-32 操作。应理解,在对 $r_{\text{Frame}} = 435$ 行数据中最后的 $r_{\text{F1}} = 15$ 行共 $q \times r_{\text{F1}} = 30840$ 个比特进行 CRC-32 操作的场景中,该 CRC-32 编码和检测操作也可以采用类似 800GZR 中的 CRC-32 操作,只是输入的比特数量

不同。

实施例 9:

考虑 800G 速率的数据帧, 一个数据帧每行包含 $q = 10280$ 比特。第一数据处理从数据帧中获取 r_{Frame} 行的数据, 总共得到 $d_{\text{in}} = q \times r_{\text{Frame}}$ 个比特。之后, 对第一数据进行 CRC 和/或插入填充比特得到第二数据。其中, 在 CRC 操作中会加入 d_{CRC} 个 CRC 校验比特, 在插入填充操作中会插入 d_{PAD} 个填充 (Pad) 比特。进而, 对第二数据进行加扰得到第三数据, 第三数据的比特数量 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CP}}$ 个加扰比特, 其中 $d_{\text{CP}} = d_{\text{CRC}} + d_{\text{PAD}}$ 。加扰后的比特数据进行包含 PCS 和 OFEC 编码、第一交织处理、OFEC 交织、16QAM 符号映射和偏振划分得到一个包含 172032 个双偏振符号, 并进行冗余为 $\frac{1}{56}$ 的 DSP 成帧得到一个包含 175104 个双偏振符号的超帧, 有:

$$d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CP}} = 336 \times k$$

所述数据处理对应的波特率为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k}\right) \times \frac{57}{8} \text{Gbaud}$, 其中 W_{bit} 为所述 800G 数据帧对应的具体比特速率值。本实施例中, 考虑 Flex0-8 数据帧, 其比特速率约为 $W_{\text{bit}} \approx 845.1481 \text{ Gbit/s}$, 波特率 W_{baud} 采用约为 131Gbaud。由于, $k \approx W_{\text{bit}} \times 4096 \times \frac{57}{8} / (131\text{G}) \approx 3362$, 结合整数 $k - 1504$ 是 8 的整数倍, 考虑整数 k 取值可为 3360 或 3368 或 3376。

对于 $k=3360$ 时, $d_{\text{scr}} = 336 \times k = 1128960$; 由于 $r_{\text{Frame}} = \lfloor 1128960/10280 \rfloor = 109$, 有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 8440$ 。此时, 波特率具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k}\right) \times \frac{57}{8} \text{Gbaud} \approx 132.072 \text{ Gbaud}$ 。由于当前产业界典型的低功耗器件对应的最高波特率约为 131Gbaud, $k=3360$ 取值不适合用于要求较低功耗应用场景。

对于 $k=3368$ 时, $d_{\text{scr}} = 336 \times k = 1131648$; 由于 $r_{\text{Frame}} = \lfloor 1131648/10280 \rfloor = 110$, 有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1131648 - 1130800 = 848$ 。此时, 波特率具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k}\right) \times \frac{57}{8} \text{Gbaud} \approx 130.871 \text{ Gbaud}$ 。

对于 $k=3376$ 时, $d_{\text{scr}} = 336 \times k = 1134336$; 由于 $r_{\text{Frame}} = \lfloor 1134336/10280 \rfloor = 110$, 有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1134336 - 1130800 = 3536$ 。此时, 波特率具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k}\right) \times \frac{57}{8} \text{Gbaud} \approx 130.871 \text{ Gbaud}$ 。

因此, 整数 k 的典型取值可为 3368 或 3376。其对应的典型参数组合如表 7 所示。

需要说明的是, 表 7 中序号 1 的参数组合 $\{r_{\text{Frame}} = 110, d_{\text{CP}} = 848, k = 3368\}$ 下, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 256, 且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.075\%$ 。表 7 中序号 2 的参数组合 $\{r_{\text{Frame}} = 110, d_{\text{CP}} = 3536, k = 3376\}$ 下, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128, 且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.313\%$ 。两种参数组合下, 序号 1 对应的 PCS 查找表具体实现复杂度较高, 但 CRC 和 PAD 冗余 OH_{CP} 较小; 序号 2 对应的 PCS 查找表具体实现复杂度较低, 但 CRC 和 PAD 冗余 OH_{CP} 较高。可根据具体应用场景选取采用序号 1 或序号 2 的参数组合进行实现。

表 7

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	FlexO-8 波特率 Gbaud
1	110	848	3368	1864	0.075%	130.871
2	110	3536	3376	1872	0.313%	130.871

需要说明的是, 对于表 7 中序号 1 的参数组合 $\{r_{\text{Frame}} = 110, d_{\text{CP}} = 848, k = 3368\}$ 下, 选取参数 $p=22$, 其具体实现可参照实施例 1。对于表 7 中序号 2 的参数组合 $\{r_{\text{Frame}} = 110, d_{\text{CP}} = 3536, k = 3376\}$ 下,

选取参数 $p=22$ ，对应的，第一数据处理从数据帧中获取 $r_{\text{Frame}} = 110$ 行的数据，总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 1130800$ 个比特。先对每 $\frac{r_{\text{Frame}}}{p} = \frac{110}{22} = 5$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 51400$ 个比特进行 CRC-32 校验，加入总共 $d_{\text{CRC}} = 22 \times 32 = 704$ 个 CRC 校验比特；然后插入 $d_{\text{PAD}} = 2832$ 个填充 (Pad) 比特，再进行加扰，得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1134336$ 个加扰比特。

5

实施例 10:

在实施例 9 基础上，考虑 Flex0-8e 数据帧，其比特速率约为 $W_{\text{bit}} \approx 804.9795 \text{ Gbit/s}$ ，波特率 W_{baud} 采用约为 131GBaud。由于， $k \approx W_{\text{bit}} \times 4096 \times \frac{57}{8} / (131\text{G}) \approx 3202$ ，结合整数 $k - 1504$ 是 8 的整数倍，考虑整数 k 取值可为 3184 或 3192 或 3200 或 3208 或 3216。

10 对于 $k=3184$ 时， $d_{\text{scr}} = 336 \times k = 1069824$ ；由于 $r_{\text{Frame}} = \lfloor 1069824/10280 \rfloor = 104$ ，有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1069824 - 1069120 = 704$ 。此时，CRC 和 PAD 冗余为 0.066%，波特率

具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k} \right) \times \frac{57}{8} \text{ Gbaud} \approx 131.842 \text{ Gbaud}$ 。

对于 $k=3192$ 时， $d_{\text{scr}} = 336 \times k = 1072512$ ；由于 $r_{\text{Frame}} = \lfloor 1072512/10280 \rfloor = 104$ ，有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1072512 - 1069120 = 3392$ 。此时，CRC 和 PAD 冗余为 0.317%，波特率

15 具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k} \right) \times \frac{57}{8} \text{ Gbaud} \approx 131.842 \text{ Gbaud}$ 。

对于 $k=3200$ 时， $d_{\text{scr}} = 336 \times k = 1075200$ ；由于 $r_{\text{Frame}} = \lfloor 1075200/10280 \rfloor = 104$ ，有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1077888 - 1069120 = 6080$ 。此时，CRC 和 PAD 冗余为 0.569%，冗余较大。

20 对于 $k=3208$ 时， $d_{\text{scr}} = 336 \times k = 1077888$ ；由于 $r_{\text{Frame}} = \lfloor 1131648/10280 \rfloor = 104$ ，有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1077888 - 1069120 = 8768$ 。此时，此时，CRC 和 PAD 冗余为 0.82%，冗余较大。

对于 $k=3216$ 时， $d_{\text{scr}} = 336 \times k = 1080576$ ；由于 $r_{\text{Frame}} = \lfloor 1080576/10280 \rfloor = 105$ ，有 $d_{\text{CP}} = d_{\text{scr}} - 10280 \times r_{\text{Frame}} = 1080576 - 1079400 = 1176$ 。此时，CRC 和 PAD 冗余为 0.109%，波特率

具体取值为 $W_{\text{baud}} = W_{\text{bit}} \times \frac{10280 \times r_{\text{Frame}} + d_{\text{CP}}}{10280 \times r_{\text{Frame}}} \times \left(\frac{4096}{k} \right) \times \frac{57}{8} \text{ Gbaud} \approx 130.587 \text{ Gbaud}$ 。

25 因此，整数 k 的典型取值可为 3184 或 3192 或 3216。其对应的典型参数组合如表 8 所示。

表 8

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	FlexO-8e 波特率 Gbaud
1	104	704	3184	1680	0.066%	131.842
2	104	3392	3192	1688	0.317%	131.842
3	105	1176	3216	1712	0.109%	130.587

需要说明的是，表 8 中序号 1 的参数组合 $\{r_{\text{Frame}} = 104, d_{\text{CP}} = 704, k = 3184\}$ 下，PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128，且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.066\%$ 。其可选取参数 $p=13$ ，对应的，第一数据处理从数据帧中获取 $r_{\text{Frame}} = 104$ 行的数据，总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} =$

30 1069120 个比特。先对每 $\frac{r_{\text{Frame}}}{p} = \frac{104}{13} = 8$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 82240$ 个比特进行 CRC-32 校验，加入总

共 $d_{\text{CRC}} = 13 \times 32 = 416$ 个 CRC 校验比特；然后插入 $d_{\text{PAD}} = 288$ 个填充 (Pad) 比特，再进行加扰，得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1069824$ 个加扰比特。

表 8 中序号 2 的参数组合 $\{r_{\text{Frame}} = 104, d_{\text{CP}} = 3392, k = 3192\}$ 下，PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 256，PCS 查找表具体实现复杂度较高，且 CRC 和 PAD 冗余 OH_{CP} 较高，因

此在 131.842GBaud 波特率下优选序号 1 的参数组合。

表 8 中序号 3 的参数组合 $\{r_{\text{Frame}} = 105, d_{\text{CP}} = 1176, k = 3216\}$ 下, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128, 且 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.109\%$ 。其对应的 PCS 查找表具体实现复杂度较低。另外, 其选取参数 $p=21$, 对应的, 第一数据处理从数据帧中获取 $r_{\text{Frame}} = 105$ 行的数据,

- 5 总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 1079400$ 个比特。先对每 $\frac{r_{\text{Frame}}}{p} = \frac{105}{21} = 5$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 51400$ 个比特进行 CRC-32 校验, 加入总共 $d_{\text{CRC}} = 21 \times 32 = 672$ 个 CRC 校验比特; 然后插入 $d_{\text{PAD}} = 504$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1080576$ 个加扰比特。

实施例 11:

- 10 考虑 800G 速率的数据帧, 一个数据帧包含 512 行, 每行包含 10280 比特。考虑表 2 中的第 10 组参数组合 $\{r_{\text{Frame}} = 110, d_{\text{CP}} = 848, k = 3368\}$ 。第一数据处理从数据帧中获取 $r_{\text{Frame}} = 110$ 行的数据, 总共得到 $d_{\text{in}} = 10280 \times r_{\text{Frame}} = 1130800$ 个比特。考虑 $p = 22$ 。

- 图 17 (a) 为本申请实施例中对数据帧进行 CRC 的一种实施方式示意图。如图 17 (a) 所示, 先对每 $\frac{r_{\text{Frame}}}{p} = \frac{110}{22} = 5$ 行的 $10280 \times \frac{r_{\text{Frame}}}{p} = 51400$ 个比特进行 CRC-32 校验, 加入总共 $d_{\text{CRC}} = 22 \times 32 = 704$ 15 个 CRC 校验比特; 然后插入 $d_{\text{PAD}} = 144$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{\text{scr}} = 10280 \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1131648$ 个加扰比特。CRC 校验和插入填充比特对应的冗余为 $\text{OH}_{\text{CP}} = \frac{d_{\text{scr}}}{d_{\text{in}}} - 1 = 0.075\%$ 。

- 如图 4 所示, 第二数据处理从第一数据处理输出的比特数据流中获取多个比特, 轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示, 将输入的 $k = 3368$ 个比特中的 $k_{\text{PCS}} = k - 1504 = 1864$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 3368$ 个比特包含 $k_{\text{PCS}} = k - 1504 = 1864$ 个比特和 1504 个不经过 PCS 处理的比特。 20

PCS 处理会采用 LUT 查找表, 每次将输入的 $k_{\text{LUT}} = 233$ 个比特进行 LUT 映射输出对应的 $n_{\text{LUT}} = 256$ 个比特, 连续操作 8 次上述 LUT 映射后将总共 $k - 1504 = 1864$ 个比特进行 LUT 映射输出对应的 2048 个比特。

- PCS 处理输出的 2048 个比特, 结合不经过 PCS 处理的 1504 个比特, 总共 $2048 + 1504 = 3552$ 个比特送入编码前的交织后得到第六比特集合, 其包含 3552 比特。第六比特集合采用图 23 (b) 所示的方阵集合表示, 且比特方阵 $I_{0,0}, I_{0,1}, I_{0,2}, I_{0,3}, I_{1,0}, I_{1,1}, I_{1,2}, I_{1,3}$ 各自的第 i_1 行第 $\left(\left\lfloor \frac{i_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列的比特 ($0 \leq i_1 < 16, 0 \leq j_1 < 8$) 以及比特方阵 $I_{0,4}, I_{0,5}, I_{1,4}, I_{1,5}$ 共 2048 比特来自经过 PCS 处理的第二比特集合。第六比特集合经过 OFEC 编码后插入 544 个 OFEC 校验比特, 得到总共 4096 个 OFEC 编码后的比特, 即第四比特集合。第四比特集合比特采用如图 11 (c) 所示的方阵集合表示, 其中比特方阵 $B_{0,0},$ 25

- 30 $B_{0,1}, B_{0,2}, B_{0,3}, B_{1,0}, B_{1,1}, B_{1,2}, B_{1,3}$ 各自的第 i_1 行第 $\left(\left\lfloor \frac{i_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 + i_1\right) \% 16$ 列比特 ($0 \leq i_1 < 16, 0 \leq j_1 < 8$) 以及比特方阵 $B_{0,4}, B_{0,5}, B_{1,4}, B_{1,5}$ 共 2048 比特来自经过 PCS 处理的第二比特集合。

- 对第四比特集合中的比特方阵 $B_{0,4}, B_{0,5}, B_{0,6}, B_{0,7}, B_{1,4}, B_{1,5}, B_{1,6}, B_{1,7}$ 进行第一交织处理得到第五比特集合中的比特方阵 $T_{0,4}, T_{0,5}, T_{0,6}, T_{0,7}, T_{1,4}, T_{1,5}, T_{1,6}, T_{1,7}$; 而比特方阵 $B_{0,0}, B_{0,1}, B_{0,2}, B_{0,3}, B_{1,0}, B_{1,1}, B_{1,2}, B_{1,3}$ 直接输出分别得到第五比特集合中的比特方阵 $T_{0,0}, T_{0,1}, T_{0,2}, T_{0,3}, T_{1,0}, T_{1,1}, T_{1,2}, T_{1,3}$ 。通过第一交织后当 $0 \leq c_1 < 64$ 时第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特。当 $64 \leq c_1 < 128$ 时, 第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_0 个比特。 $c_1 = (c \% \theta + (r_1 + 4 \times \Delta) \% \beta) \% \theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$ 。其中, $64 \leq c_0 < 128, 64 \leq c_1 < 128, \theta$ 是 4 的整数倍, 且 $m_0 \times 4$ 能被 θ 整除, Δ 为 35

整数, β 是 4 的整数倍, $c = \left\lfloor \frac{c_0 \% 32}{2} \right\rfloor \times 4 + c_0 \% 2 + \left(1 - \left\lfloor \frac{c_0}{32} \right\rfloor \% 2\right) \times 2 + \left\lfloor \frac{c_0}{64} \right\rfloor \times 64$ 。

这里 $r_0 = r_1$ 。采用编码前交织和第一交织处理, 得到交织后的第五比特集合中任一比特方阵 T_{i_0, j_0} ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特是来自经过 PCS 处理得到的第二比特集合 (2048 个比特) 中的 1 个比特, 其中, $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

- 5 $d_{scr} = 1131648$ 个加扰比特经 PCS 处理、编码前交织、OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍, 有利于硬件实现。OFEC 交织输出的 1376256 个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号, 并进行 DSP 成帧, 即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS, 得到一个包含 175104 个双偏振符号的超帧, 然后进行发送。

10 给定数据帧的速率为 W_{bit} Gbit/s, 对应的波特率为 $W_{baud} = W_{bit} \times (10280 \times r_{Frame} + d_{CP}) / (10280 \times r_{Frame}) \times (4096/k) \times (57/56) / 8$ Gbaud。对于 Flex0-8e、Flex0-8 数据帧类型, 其对应波特率分别为 124.7 Gbaud 和 130.9 Gbaud。

15 实施例 12:

考虑 800G 速率的数据帧, 一个数据帧包含 512 行, 每行包含 10280 比特。考虑表 3 中的第 3 组参数组合 $\{r_{Frame} = 261, d_{CP} = 1560, k = 3196\}$ 。第一数据处理从数据帧中获取 $r_{Frame} = 261$ 行的数据, 总共得到 $d_{in} = 10280 \times r_{Frame} = 2683080$ 个比特。

- 考虑 $p = 29$ 。先对每 $\frac{r_{Frame}}{p} = \frac{261}{29} = 9$ 行的 $10280 \times \frac{r_{Frame}}{p} = 92520$ 个比特进行 CRC-32 校验, 加入总
20 共 $d_{CRC} = 29 \times 32 = 928$ 个 CRC 校验比特; 然后插入 $d_{PAD} = 632$ 个填充 (Pad) 比特, 再进行加扰, 得到 $d_{scr} = 10280 \times r_{Frame} + d_{CRC} + d_{PAD} = 2684640$ 比特。CRC 校验和插入填充比特对应的冗余为 $OH_{CP} = \frac{d_{scr}}{d_{in}} - 1 = 0.058\%$ 。

- 如图 4 所示, 第二数据处理从第一数据处理输出的比特数据流中获取多个比特, 轮询送入 $L = 4$ 个 PCS 和 OFEC 编码进行 PCS 处理和 OFEC 编码。如图 5 所示, PCS 和 OFEC 编码将输入的 $k = 3196$ 个比特中的 $k_{PCS} = k - 1504 = 1692$ 个比特送入 PCS 处理得到 2048 个比特。 $k = 3196$ 个比特包含 $k_{PCS} = k - 1504 = 1692$ 个比特和 1504 个比特。

PCS 处理会采用 LUT 查找表, 每次将输入的 $k_{LUT} = 423$ 个比特进行 LUT 映射输出对应的 $n_{LUT} = 512$ 个比特, 连续操作 4 次上述 LUT 映射后将总共 $k - 1504 = 1692$ 个比特进行 LUT 映射输出对应的 2048 个比特。

- 30 PCS 处理输出的 2048 个比特, 结合不经过 PCS 处理的 1504 个比特, 总共 $2048 + 1504 = 3552$ 个比特送入编码前交织处理后得到第六比特集合, 其包含 3552 比特。第六比特集合采用图 23(b) 所示的方阵集合表示, 且比特方阵 $I_{0,0}, I_{0,1}, I_{0,2}, I_{0,3}, I_{0,4}, I_{1,0}, I_{1,1}, I_{1,2}, I_{1,3}, I_{1,4}$ 各自的第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 - i_1 \% 2\right)$ 列的比特 ($0 \leq i_1 < 16, 0 \leq j_1 < 8$)、比特方阵 $I_{0,5}, I_{1,5}$ 、比特方阵 $I_{0,6}$ 的第 0 列到到 7 列、比特方阵 $I_{1,6}$ 的第 0 列到到 7 列共 2048 比特来自经过 PCS 处理的第二比特集合。第六比特集合经过 OFEC 编码后插入 544 个 OFEC 校验比特, 得到总共 4096 个 OFEC 编码后的比特, 即第四比特集合。第四比特集合比特采用如图 11(c) 所示的方阵集合表示, 其中比特方阵 $B_{0,0}, B_{0,1}, B_{0,2}, B_{0,3}, B_{0,4}, B_{1,0}, B_{1,1}, B_{1,2}, B_{1,3}, B_{1,4}$ 各自的第 i_1 行第 $\left(\left\lfloor \frac{j_1}{2} \right\rfloor \times 4 + j_1 \% 2 + 2 + i_1\right) \% 16$ 列比特 ($0 \leq i_1 < 16, 0 \leq j_1 < 8$)、比特方阵 $B_{0,5}, B_{1,5}$ 、比特方阵 $B_{0,6}$ 中阴影部分比特, 以及比特方阵 $B_{1,6}$ 阴影部分比特共 2048 比特来自经过 PCS 处理的第二比特集合。

对第四比特集合中的比特方阵 $B_{0,5}$ 、 $B_{0,6}$ 、 $B_{0,7}$ 、 $B_{1,5}$ 、 $B_{1,6}$ 、 $B_{1,7}$ 进行第一交织处理得到第五比特集合中的比特方阵 $T_{0,5}$ 、 $T_{0,6}$ 、 $T_{0,7}$ 、 $T_{1,5}$ 、 $T_{1,6}$ 、 $T_{1,7}$ 。而比特方阵 $B_{0,0}$ 、 $B_{0,1}$ 、 $B_{0,2}$ 、 $B_{0,3}$ 、 $B_{0,4}$ 、 $B_{1,0}$ 、 $B_{1,1}$ 、 $B_{1,2}$ 、 $B_{1,3}$ 、 $B_{1,4}$ 直接输出分别得到第五比特集合中的比特方阵 $T_{0,0}$ 、 $T_{0,1}$ 、 $T_{0,2}$ 、 $T_{0,3}$ 、 $T_{0,4}$ 、 $T_{1,0}$ 、 $T_{1,1}$ 、 $T_{1,2}$ 、 $T_{1,3}$ 、 $T_{1,4}$ ；通过第一交织后。当 $0 \leq c_1 < 80$ 时，第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于第四比特集合中第 r_0 行的 128 个比特中第 c_1 个比特。当 $80 \leq c_1 < 128$ 时，第五比特集合中第 r_1 行的 128 个比特中第 c_1 个比特来自于所述第四比特集合中第 r_0 行的 128 个比特中第 $80 + c_0$ 个比特。

$c_1 = 80 + (c_0\theta + (r_1 + 2 + 4 \times \Delta)\beta)\theta + \left\lfloor \frac{c}{\theta} \right\rfloor \times \theta$ 。其中， $0 \leq c_0 < 48$ ， $80 \leq c_1 < 128$ ， θ 是 4 的整数

倍，且 48 能被 θ 整除， Δ 为整数， β 是 4 的整数， $c = \left\lfloor \frac{c_2\%24}{2} \right\rfloor \times 4 + c_2\%2 + \left\lfloor \frac{c_2}{24} \right\rfloor \times 2$ ， $c_2 = (r_0\%16)^{\wedge}c_0$ 。

这里 $r_0 = r_1$ 。采用编码前交织和第一交织处理，得到交织后的第五比特集合中任一比特方阵 T_{i_0,j_0} ($0 \leq i_0 < 2$ 且 $0 \leq j_0 < 8$) 中第 i_2 行第 $\left(\left\lfloor \frac{j_2}{2} \right\rfloor \times 4 + j_2\%2 + i_2 + 2\right)\%16$ 列的比特是来自经过 PCS 处理得到的第二比特集合 (2048 个比特) 中的 1 个比特，其中， $0 \leq i_2 < 16$ 且 $0 \leq j_2 < 8$ 。

$d_{scr} = 1131648$ 个加扰比特经 PCS 处理、编码前交织、OFEC 编码以及第一交织处理得到 $\frac{d_{scr}}{4 \times k} \times (4 \times 4096) = 1376256$ 个经第一交织处理的比特。其送入 OFEC 交织器进行“方阵内交织”和“方阵间交织”。OFEC 交织器的分组大小为 172032 比特。经第一交织处理的比特长度是 172032 的整数倍，有利于硬件实现。OFEC 交织输出的 1376256 个比特送入 DP-16QAM 符号映射和偏振划分得到 172032 个双偏振符号，并进行 DSP 成帧，即对 X 偏振和 Y 偏振符号插入 FAW、TS、RES 和 PS，得到一个包含 175104 个双偏振符号的超帧，然后进行发送。

给定数据帧的速率为 W_{bit} Gbit/s，对应的波特率为 $W_{baud} = W_{bit} \times (10280 \times r_{Frame} + d_{CP}) / (10280 \times r_{Frame}) \times (4096/k) \times (57/56) / 8$ Gbaud。对于 FlexO-8e、FlexO-8 数据帧类型，其对应波特率分别为 124.7 Gbaud 和 130.9 Gbaud。

需要说明的是，上述表 2 或表 3 给出了在数据帧结构采用每行包含 $q = 10280$ 个比特下的一些参数组合，包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 以及采用 FlexO-6e、FlexO-6、FlexO-8e、FlexO-8 数据帧类型下对应的波特率。下面考虑数据帧结构采用每行包含 $q = 2056$ 个比特的情况下一些可能的参数组合。

对于 FlexO-8 数据帧类型，下面表 9 给出了在数据帧结构采用每行包含 $q = 2056$ 个比特下的一些参数组合，包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 、采用 FlexO-8 数据帧类型下对应的波特率。

表 9

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	FlexO-8 波特率 Gbaud
1	546	1008	3344	1840	0.09%	131.8297
2	546	3696	3352	1848	0.33%	131.8297
3	547	1640	3352	1848	0.15%	131.5887
4	547	4328	3360	1856	0.38%	131.5887
5	548	2272	3360	1856	0.20%	131.3485
6	548	4960	3368	1864	0.44%	131.3485

7	549	216	3360	1856	0.02%	131.1093
8	549	2904	3368	1864	0.26%	131.1093
9	549	5592	3376	1872	0.50%	131.1093
10	550	848	3368	1864	0.08%	130.8709
11	550	3536	3376	1872	0.31%	130.8709
12	551	1480	3376	1872	0.13%	130.6334
13	551	4168	3384	1880	0.37%	130.6334
14	552	2112	3384	1880	0.19%	130.3967
15	552	4800	3392	1888	0.42%	130.3967
16	553	56	3384	1880	0.00%	130.1609
17	553	2744	3392	1888	0.24%	130.1609
18	553	5432	3400	1896	0.48%	130.1609
19	554	688	3392	1888	0.06%	129.9260
20	554	3376	3400	1896	0.30%	129.9260
21	555	1320	3400	1896	0.12%	129.6919
22	555	4008	3408	1904	0.35%	129.6919
23	556	1952	3408	1904	0.17%	129.4586
24	556	4640	3416	1912	0.41%	129.4586
25	557	2584	3416	1912	0.23%	129.2262
26	557	5272	3424	1920	0.46%	129.2262
27	558	528	3416	1912	0.05%	128.9946
28	558	3216	3424	1920	0.28%	128.9946
29	559	1160	3424	1920	0.10%	128.7639
30	559	3848	3432	1928	0.33%	128.7639
31	560	1792	3432	1928	0.16%	128.5339
32	560	4480	3440	1936	0.39%	128.5339
33	561	2424	3440	1936	0.21%	128.3048
34	561	5112	3448	1944	0.44%	128.3048
35	562	368	3440	1936	0.03%	128.0765
36	562	3056	3448	1944	0.26%	128.0765
37	562	5744	3456	1952	0.50%	128.0765

对于 Flex0-8e 数据帧类型下，下面表 10 给出了在数据帧结构采用每行包含 $q = 2056$ 个比特下的一些参数组合，包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 、采用 Flex0-8e 数据帧类型下对应的波特率。

5

表 10

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	FlexO-8e 波特率 Gbaud
1	520	704	3184	1680	0.07%	131.8422
2	520	3392	3192	1688	0.32%	131.8422
3	520	6080	3200	1696	0.57%	131.8422
4	521	1336	3192	1688	0.12%	131.5891

5	521	4024	3200	1696	0.38%	131.5891
6	522	1968	3200	1696	0.18%	131.3371
7	522	4656	3208	1704	0.43%	131.3371
8	523	2600	3208	1704	0.24%	131.0859
9	523	5288	3216	1712	0.49%	131.0859
10	524	544	3208	1704	0.05%	130.8358
11	524	3232	3216	1712	0.30%	130.8358
12	525	1176	3216	1712	0.11%	130.5866
13	525	3864	3224	1720	0.36%	130.5866
14	526	1808	3224	1720	0.17%	130.3383
15	526	4496	3232	1728	0.42%	130.3383
16	527	2440	3232	1728	0.23%	130.0910
17	527	5128	3240	1736	0.47%	130.0910
18	528	384	3232	1728	0.04%	129.8446
19	528	3072	3240	1736	0.28%	129.8446
20	528	5760	3248	1744	0.53%	129.8446
21	529	1016	3240	1736	0.09%	129.5991
22	529	3704	3248	1744	0.34%	129.5991
23	530	1648	3248	1744	0.15%	129.3546
24	530	4336	3256	1752	0.40%	129.3546
25	531	2280	3256	1752	0.21%	129.1110
26	531	4968	3264	1760	0.46%	129.1110
27	532	2912	3264	1760	0.27%	128.8683
28	532	224	3256	1752	0.02%	128.8683
29	533	856	3264	1760	0.08%	128.6265
30	533	3544	3272	1768	0.32%	128.6265
31	534	1488	3272	1768	0.14%	128.3857
32	534	4176	3280	1776	0.38%	128.3857
33	535	2120	3280	1776	0.19%	128.1457
34	535	4808	3288	1784	0.44%	128.1457

实施例 13:

在一些具体应用场景中，当 Flex0-8 波特率和 Flex0-8e 波特率的数值相近时，可以采用相同或相似的器件实现。结合上述表 9 和表 10，可看出采用 Flex0-8 波特率 131.5887Gbaud 和 Flex0-8e 波特率 131.5891Gbaud 时，相差数值除以 Flex0-8 或 Flex0-8e 波特率数值仅为 3.7237E-6，对应 3.7237ppm，相差数值较小。下表 11 给出了具体参数组合。

5

表 11

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	数据帧类型
1	547	1640	3352	1848	0.15%	131.5887	Flex0-8
2	547	4328	3360	1856	0.38%	131.5887	
3	521	1336	3192	1688	0.12%	131.5891	Flex0-8e
4	521	4024	3200	1696	0.38%	131.5891	

在一些具体应用中,对于 Flex0-8 数据帧类型,考虑参数组合 $\{r_{\text{Frame}} = 547, d_{\text{CP}} = 1640, k = 3352\}$ 。第一数据处理从数据帧中获取 $r_{\text{Frame}} = 547$ 行的数据, 总共得到 $d_{\text{in}} = q \times r_{\text{Frame}} = 2056 \times 547 = 1124632$ 个比特, 选取参数 $p=28$ 。考虑整数 $r_{\text{F0}} = \left\lfloor \frac{r_{\text{Frame}}}{p} \right\rfloor + 1 = 20$, 整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 547 - 20 \times 27 = 7$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 547$ 行数据, 对其中在先的 $r_{\text{F0}} \times (p - 1) = 540$ 行中的每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特得到 $32 \times (p - 1) = 864$ 个比特的 CRC 校验比特, 再对其中最后的 $r_{\text{F1}} = 7$ 行共 $q \times r_{\text{F1}} = 14392$ 个比特进行 CRC-32 添加长度 32 比特的校验比特, 总共得到 $d_{\text{CRC}} = 32 \times p = 896$ 个比特的 CRC 校验比特。然后, 插入 $d_{\text{PAD}} = 744$ 个填充 (Pad) 比特, 再进行加扰得到 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1126272$ 个加扰比特。需要说明的是, 上述 CRC-32 编码和检测操作可直接采用现有 800GZR 中的 CRC-32 操作, 区别在于现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。对应的, PCS 处理对应的输入比特数量 $k_{\text{PCS}} = k - 1504 = 1848$ 。PCS 处理通常会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。这里, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 256, 其对应的 PCS 查找表具体实现复杂度较高; 但 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.15\%$ 较低。

在另一些具体应用中, 对于 Flex0-8 数据帧类型, 考虑参数组合 $\{r_{\text{Frame}} = 547, d_{\text{CP}} = 4328, k = 3360\}$ 。第一数据处理从数据帧中获取 $r_{\text{Frame}} = 547$ 行的数据, 总共得到 $d_{\text{in}} = q \times r_{\text{Frame}} = 2056 \times 547 = 1124632$ 个比特, 选取参数 $p=28$ 。考虑整数 $r_{\text{F0}} = \left\lfloor \frac{r_{\text{Frame}}}{p} \right\rfloor + 1 = 20$, 整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 547 - 20 \times 27 = 7$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 547$ 行数据, 对其中在先的 $r_{\text{F0}} \times (p - 1) = 540$ 行中的每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特得到 $32 \times (p - 1) = 864$ 个比特的 CRC 校验比特, 再对最后的 $r_{\text{F1}} = 7$ 行共 $q \times r_{\text{F1}} = 14392$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特, 总共得到 $d_{\text{CRC}} = 32 \times p = 896$ 个比特的 CRC 校验比特。然后插入 $d_{\text{PAD}} = 3432$ 个填充 (Pad) 比特, 再进行加扰得到 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1128960$ 个加扰比特。需要说明的是, 上述 CRC-32 编码和检测操作可直接采用现有 800GZR 中的 CRC-32 操作, 区别在于现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。对应的, PCS 处理对应的输入比特数量 $k_{\text{PCS}} = k - 1504 = 1856$ 。PCS 处理通常会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。这里, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128, 其对应的 PCS 查找表具体实现复杂度较低; 但 CRC 和 PAD 冗余 $\text{OH}_{\text{CP}} = 0.38\%$ 较高。需要说明的是, 在本实施例中, PCS 处理对应的输入比特数量 $k_{\text{PCS}} = k - 1504 = 1856$, PCS 处理对应的输出总比特数是 2048, 且采用查找表 LUT 将所述 1856 比特输入比特中每 116 个比特进行处理得到 128 个比特。在一些具体应用中, 所述查找表 LUT 可采用包含多个子查找表来实现, 所述子查找表的输入比特数可以不相同, 并且输出比特数也可以不相同。例如所述查找表采用 12 个子查找表实现, 其中 4 个子查找表将每 9 个输入比特进行处理得到 10 个输出比特, 另外 8 个子查找表将每 10 个输入比特进行处理得到 11 个输出比特。本申请中, 对所述查找表具体实现不做限定。

在一些具体应用中, 对于 Flex0-8e 数据帧类型下, 考虑参数组合 $\{r_{\text{Frame}} = 521, d_{\text{CP}} = 1336, k = 3192\}$ 。第一数据处理从数据帧中获取 $r_{\text{Frame}} = 521$ 行的数据, 总共得到 $d_{\text{in}} = q \times r_{\text{Frame}} = 2056 \times 521 = 1071176$ 个比特, 选取参数 $p=27$ 。考虑整数 $r_{\text{F0}} = \left\lfloor \frac{r_{\text{Frame}}}{p} \right\rfloor + 1 = 20$, 整数 $r_{\text{F1}} = r_{\text{Frame}} - r_{\text{F0}} \times (p - 1) = 521 - 20 \times 26 = 1$ 。针对从数据帧中获取的 $r_{\text{Frame}} = 521$ 行数据, 对其中在先的 $r_{\text{F0}} \times (p - 1) = 520$ 行中的每 $r_{\text{F0}} = 20$ 行共 $q \times r_{\text{F0}} = 41120$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特得到 $32 \times (p - 1) = 832$ 个比特的 CRC 校验比特, 再对最后的 $r_{\text{F1}} = 1$ 行共 $q \times r_{\text{F1}} = 2056$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特, 总共得到 $d_{\text{CRC}} = 32 \times p = 864$ 个比特的 CRC 校验比特。然后, 插入 $d_{\text{PAD}} = 472$ 个填充 (Pad) 比特, 再进行加扰得到 $d_{\text{scr}} = q \times r_{\text{Frame}} + d_{\text{CRC}} + d_{\text{PAD}} = 1072512$ 个加扰比特。需要说明的是, 上述 CRC-32 编码和检测操作可直接采用现有 800GZR 中的 CRC-32 操作, 区别在于

现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。对应的, PCS 处理对应的输入比特数量 $k_{PCS} = k - 1504 = 1688$ 。PCS 处理通常会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。这里, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 256, 其对应的 PCS 查找表具体实现复杂度较高; 但 CRC 和 PAD 冗余 $OH_{CP} = 0.12\%$ 较低。

- 5 在另一些具体应用中, 对于 Flex0-8e 数据帧类型下, 考虑参数组合 $\{r_{Frame} = 521, d_{CP} = 4024, k = 3200\}$ 。第一数据处理从数据帧中获取 $r_{Frame} = 521$ 行的数据, 总共得到 $d_{in} = q \times r_{Frame} = 2056 \times 521 = 1071176$ 个比特, 选取参数 $p=27$ 。考虑整数 $r_{F0} = \left\lfloor \frac{r_{Frame}}{p} \right\rfloor + 1 = 20$, 整数 $r_{F1} = r_{Frame} - r_{F0} \times (p - 1) = 521 - 20 \times 26 = 1$ 。针对从数据帧中获取的 $r_{Frame} = 521$ 行数据, 对其中在先的 $r_{F0} \times (p - 1) = 520$ 行中的每 $r_{F0} = 20$ 行共 $q \times r_{F0} = 41120$ 个比特进行 CRC-32 添加长度 32 为比特的校验比特得到 $32 \times (p - 1) = 832$ 个比特的 CRC 校验比特, 再对最后的 $r_{F1} = 1$ 行共 $q \times r_{F1} = 2056$ 个比特进行 CRC-32 添加长度为 32 比特的校验比特, 总共得到 $d_{CRC} = 32 \times p = 864$ 个比特的 CRC 校验比特。然后, 插入 $d_{PAD} = 3160$ 个填充 (Pad) 比特, 再进行加扰得到 $d_{scr} = q \times r_{Frame} + d_{CRC} + d_{PAD} = 1075200$ 个加扰比特。需要说明的是, 上述 CRC-32 编码和检测操作可直接采用现有 800GZR 中的 CRC-32 操作, 区别在于现有 800GZR 中对 4 行 10280 列的共 $4 \times 10280 = 41120$ 比特数据进行 CRC-32 操作。对应的, PCS 处理对应的输入比特数量 $k_{PCS} = k - 1504 = 1696$ 。PCS 处理通常会采用查找表 (lookup table, LUT) 将输入的 k_{LUT} 个比特进行映射输出对应的 n_{LUT} 个比特。这里, PCS 处理中所采用的查找表输出比特数量 n_{LUT} 最小取值为 128, 其对应的 PCS 查找表具体实现复杂度较低; 但 CRC 和 PAD 冗余 $OH_{CP} = 0.38\%$ 较高。

实施例 14:

- 20 在一些具体应用场景中, 考虑采用无损 (hitless) 速率切换, 通常考虑不同速率采用相同波特率。比如 Flex0-8 数据帧类型下对应的波特率和 Flex0-6 数据帧类型下对应的波特率相同, 比如 Flex0-8e 数据帧类型下对应的波特率和 Flex0-6e 数据帧类型下对应的波特率相同, 以支撑无损速率切换。为了便于整体方案硬件实现, 要求 Flex0-8 数据帧类型下的 r_{Frame} 取值 $r_{Frame-8}$ 为 8 的整数倍, Flex0-6 数据帧类型下的 r_{Frame} 取值 $r_{Frame-6}$ 为 6 的整数倍, 且 $\frac{r_{Frame-8}}{8} = \frac{r_{Frame-6}}{6}$ 。
- 25 表 12 给出了 Flex0-8 和 Flex0-6 数据帧类型下对应的波特率均为相同数值 130.3967Gbaud 的参数组合, 包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 以及波特率。可以看出, Flex0-8 数据帧类型下的 r_{Frame} 取值 $r_{Frame-8}$ 为 8 的整数倍, Flex0-6 数据帧类型下的 r_{Frame} 取值 $r_{Frame-6}$ 为 6 的整数倍, 且 $\frac{r_{Frame-8}}{8} = \frac{r_{Frame-6}}{6}$, 便于支撑无损 (hitless) 速率切换实现。

表 12

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	备注
1	552	2112	3384	1880	0.19%	130.3967	Flex0-8
2	552	4800	3392	1888	0.42%	130.3967	
3	414	912	2536	1032	0.11%	130.3967	Flex0-6
4	414	3600	2544	1040	0.42%	130.3967	

表 13 给出了 Flex0-8 和 Flex0-6 数据帧类型下对应的波特率均为相同数值 128.5339Gbaud 的参数组合, 包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 以及波特率。可以看出, Flex0-8 数据帧类型下的 r_{Frame} 取值 $r_{Frame-8}$ 为 8 的整数倍, Flex0-6 数据帧类型下的

r_{Frame} 取值 $r_{Frame-6}$ 为6的整数倍,且 $\frac{r_{Frame-8}}{8} = \frac{r_{Frame-6}}{6}$,便于支撑无损(hitless)速率切换实现。

表 13

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	备注
1	560	1792	3432	1928	0.16%	128.5339	Flex0-8
2	560	4480	3440	1936	0.39%	128.5339	
3	420	2016	2576	1072	0.23%	128.5339	Flex0-6
4	420	4704	2584	1080	0.54%	128.5339	

实施例 15:

在实施例 14 基础上,考虑 Flex0-8e 数据帧和 Flex0-6e 数据帧。表 14 给出了 Flex0-8e 和 Flex0-6e 数据帧类型下对应的波特率均为相同数值 131.8422Gbaud 的参数组合,包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 以及波特率。可以看出, Flex0-8e 数据帧类型下的 r_{Frame} 取值 $r_{Frame-8e}$ 为8的整数倍, Flex0-6e 数据帧类型下的 r_{Frame} 取值 $r_{Frame-6e}$ 为6的整数倍,且 $\frac{r_{Frame-8e}}{8} = \frac{r_{Frame-6e}}{6}$,便于支撑无损(hitless)速率切换实现。

10

表 14

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	备注
1	520	704	3184	1680	0.07%	131.8422	Flex0-8e
2	520	3392	3192	1688	0.32%	131.8422	
3	520	6080	3200	1696	0.57%	131.8422	
4	390	1872	2392	888	0.23%	131.8422	Flex0-6e
5	390	4560	2400	896	0.57%	131.8422	

表 15 给出了 Flex0-8e 和 Flex0-6e 数据帧类型下对应的波特率均为相同数值 129.8446Gbaud 的参数组合,包括每次从数据帧中获取数据行数 r_{Frame} 、CRC 校验和插入填充比特的总比特数 d_{CP} 、PCS 处理和 OFEC 编码输入比特数 k 、PCS 处理的输入比特数 k_{PCS} 、CRC 校验和插入填充比特对应的冗余 OH_{CP} 以及波特率。可以看出, Flex0-8e 数据帧类型下的 r_{Frame} 取值 $r_{Frame-8e}$ 为8的整数倍, Flex0-6e 数据帧类型下的 r_{Frame} 取值 $r_{Frame-6e}$ 为6的整数倍,且 $\frac{r_{Frame-8e}}{8} = \frac{r_{Frame-6e}}{6}$,便于支撑无损(hitless)速率切换实现。

15

表 15

序号	r_{Frame}	d_{CP}	k	k_{PCS}	OH_{CP}	波特率 Gbaud	备注
1	528	384	3232	1728	0.04%	129.8446	Flex0-8e
2	528	3072	3240	1736	0.28%	129.8446	
3	528	5760	3248	1744	0.53%	129.8446	
4	396	288	2424	920	0.04%	129.8446	Flex0-6e
5	396	2976	2432	928	0.37%	129.8446	

图 18 为本申请实施例中第三数据处理的一个实施例示意图。如图 18 所示,区别于上述图 4 所示实施例介绍的第二数据处理,第三数据处理中在 PCS 和 FEC 编码后只需要进行一种第三交织处理,而不用像第二数据处理中执行第一交织处理和第二交织处理。其中,第三数据处理中的第三交织处理区别于上述的第一交织处理和第二交织处理,具体可以是一种包含“方阵内交织”和“优化方阵间交织”的 FEC 交织。下面对第三数据处理进行详细介绍。

需要说明的是,第三数据处理中的 PCS 和 FEC 编码与第二数据处理中的 PCS 和 FEC 编码是类似的操作,具体可以参考上述图 4 所示实施例的相关介绍,此处不再赘述。还需要说明的是,第三数据处理中的 DSP 成帧与第二数据处理中的 DSP 成帧是类似的操作,具体可以参考上述图 4 所示实施例的相关介绍,此处不再赘述。

如图 18 所示,每两个 FEC 编码器的输出送入一个第三交织器进行处理,第三交织处理包含“方阵内交织”和“优化方阵间交织”共 2 个操作。其中,此处的“方阵内交织”操作与第二数据处理中的“方阵内交织”操作类似,具体可以参考图 14 所示实施例的相关介绍,其对每个 16 行 16 列的比特方阵进行比特交织。

作为一个示例,输入到第三交织器的两路比特集合分别记为比特集合 0 和比特集合 1,比特集合 0 为第 0 路 PCS 和 FEC 编码后的输出,比特集合 1 为第 1 路 PCS 和 FEC 编码后的输出。比特集合 0 包括 m_0 个比特方阵 0,每个比特方阵 0 包括分布为 16 行 16 列的共 256 个比特。比特集合 1 包括 m_0 个比特方阵 1,每个比特方阵 1 包括分布为 16 行 16 列的共 256 个比特。第三交织器包括交织缓存,“优化方阵间交织”的具体操作包括将 21 个比特集合 0 和 21 个比特集合 1 输入交织缓存,并按照一定规则从交织缓存中读取比特以进行符号映射和偏振划分。

其中,交织缓存包括分布为 84 行 m_1 列的共 $84 \times m_1$ 个缓存单元, $m_0 = 2 \times m_1$,每个缓存单元用于缓存 16 行 16 列共 256 个比特。 $2 \times m_1$ 个所述比特方阵 0 输入交织缓存的偶数行, $42 \times m_1$ 个所述比特方阵 1 输入交织缓存的奇数行。交织缓存包括第一缓存子集和第二缓存子集,所述第一缓存子集包括 $42 \times m_1$ 个所述缓存单元,所述第二缓存子集包括 $42 \times m_1$ 个所述缓存单元。输入所述第一缓存子集的比特来自第 0 路 PCS 和 FEC 编码中 PCS 处理后输出的比特和第 1 路 PCS 和 FEC 编码中 PCS 处理后输出的比特。输入所述第二缓存子集的比特来自第 0 路 PCS 和 FEC 编码中未经 PCS 处理的比特和 FEC 校验比特以及第 1 路 PCS 和 FEC 编码中未经 PCS 处理的比特和 FEC 校验比特。

进而,对所述交织缓存中每 8 个比特进行符号映射和偏振划分得到 1 个双偏振 DP-16QAM 符号,所述 DP-16QAM 符号在目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个幅度位比特来自所述第一缓存子集的第一缓存单元中的一列,所述 DP-16QAM 符号在所述目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个符号位比特来自所述第二缓存子集的第二缓存单元中的一列。更具体的,上述参与符号映射和偏振划分的每 8 个比特中的其中 4 个比特来自所述第一缓存子集的第一缓存单元中的一列,所述 8 个比特中的另外 4 个比特来自所述第二缓存子集的第二缓存单元中的一列。

在一种可能的实施方式中,本申请具体以 $m_0 = 16$ 为例进行介绍,即交织缓存包括分布为 84 行 8 列共 672 个缓存单元,一共可缓存 172032 个比特,第一缓存子集包括 336 个缓存单元,第二缓存子集包括 336 个缓存单元。

图 19 为本申请实施例中交织缓存的一种分布示意图。应理解,在比特方阵按上述图 11(a) 所示分布方式的基础,设计了如图 19 所示的交织缓存分布。作为一个示例,在图 19 中 $m_0 = 16$,所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 2 列和第 3 列中的共 336 个缓存单元,所述第二缓存子集包括所述交织缓存中第 4 列、第 5 列、第 6 列和第 7 列中的共 336 个缓存单元。

作为一个示例, $m_0 = 12$,所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列和第 2 列中的共 252 个缓存单元,所述第二缓存子集包括所述交织缓存中第 3 列、第 4 列和第 5 列中的共 252 个缓存单元。

图 20 为本申请实施例中交织缓存的另一种分布示意图。应理解,在比特方阵按上述图 11(b) 所示分布方式的基础,设计了如图 20 所示的交织缓存分布。作为一个示例,在图 20 中 $m_0 = 16$,所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 4 列和第 5 列中的共 336 个缓存单元,所述第二缓存子集包括所述交织缓存中第 2 列、第 3 列、第 6 列和第 7 列中的共 336 个缓存单元。

通过上面的介绍可知，本申请提供的第三数据处理相对于上述的第二数据处理来说，第三交织处理与第二交织处理所用的缓存大小相近或相同。相当于第三交织处理替换了上述第一方面中的第一交织处理和第二交织处理，达到了与第一方面类似的效果，使得经过 PCS 处理得到的 0 和 1 不等概的比特可以映射到调制符号的幅度位比特，实现了在保持星座点位置不变下改变星座点出现的概率，使其非均匀分布，提升了整体性能，以满足未来更长传输距离需求，并且，第三数据处理具有更低的复杂度。

下面介绍本申请实施例提供的数据处理装置。

图 25 为本申请实施例中数据处理装置的第一种结构示意图。如图 25 所示，该数据处理装置包括：PCS 单元 101、FEC 编码单元 102 和第一交织处理单元 103。PCS 单元 101 用于执行上述图 5 所示实施例介绍的 PCS 处理。FEC 编码单元 102 用于执行上述图 5 所示实施例介绍的 FEC 编码。第一交织处理单元 103 用于执行上述图 4 所示实施例介绍的第一交织处理。PCS 单元 101、FEC 编码单元 102 和第一交织处理单元 103 的具体操作可以参考上述图 4 和图 5 所示实施例的相关介绍，此处不再赘述。

在一些可能的实施方式中，图 25 所示的数据处理装置还可以包括：第二交织处理单元、符号映射单元、偏振划分单元和 DSP 成帧单元。第二交织处理单元、符号映射单元、偏振划分单元和 DSP 成帧单元分别用于执行图 4 所示实施例介绍的第二数据处理中的第二交织处理、符号映射、偏振划分和 DSP 成帧，具体操作可以参考上述图 4 所示实施例的相关介绍，此处不再赘述。

在一些可能的实施方式中，图 25 所示的数据处理装置还可以包括编码前交织单元，编码前交织单元用于执行图 21 所示实施例介绍的编码前交织处理，具体操作可以参考上述图 21 所示实施例的相关介绍，此处不再赘述。

图 26 为本申请实施例中数据处理装置的第二种结构示意图。如图 26 所示，该数据处理装置包括：第一处理单元 201、第二处理单元 202、加扰单元 203、第三处理单元 204、PCS 单元 205 和 FEC 编码单元 206。第一处理单元 201 用于从数据帧中获取包括 r 行 q 列个比特的第一数据。第二处理单元 202 用于对第一数据执行上述第一数据处理中的 CRC 校验和/或插入填充比特得到第二数据。加扰单元 203 用于对第二数据执行上述第一数据处理中的加扰操作得到第三数据。第三处理单元 204 用于从第三数据中获取 k 个比特。PCS 单元 205 用于执行上述图 5 所示实施例介绍的 PCS 处理。FEC 编码单元 206 用于执行上述图 5 所示实施例介绍的 FEC 编码。应理解，该数据处理装置的具体操作可以参考上述图 3-图 5 所示实施例的相关介绍，此处不再赘述。

在一些可能的实施方式中，图 26 所示的数据处理装置还可以包括：第一交织单元、第二交织单元、符号映射单元、偏振划分单元和 DSP 成帧单元。第一交织单元、第二交织处理单元、符号映射单元、偏振划分单元和 DSP 成帧单元分别用于执行图 4 所示实施例介绍的第二数据处理中的第一交织处理、第二交织处理、符号映射、偏振划分和 DSP 成帧，具体操作可以参考上述图 4 所示实施例的相关介绍，此处不再赘述。

在一些可能的实施方式中，图 26 所示的数据处理装置还可以包括编码前交织单元，编码前交织单元用于执行图 21 所示实施例介绍的编码前交织处理，具体操作可以参考上述图 21 所示实施例的相关介绍，此处不再赘述。

图 27 为本申请实施例中数据处理装置的第三种结构示意图。如图 27 所示，该数据处理装置包括：第一 PCS 单元 301、第一 FEC 编码单元 302、第二 PCS 单元 303、第二 FEC 单元 304、交织处理单元 305、符号映射单元 306 和偏振划分单元 307。第一 PCS 单元 301 和第二 PCS 单元 303 都用于执行上述图 5 所示实施例介绍的 PCS 处理。第一 FEC 编码单元 302 和第二 FEC 单元 304 都用于执行上述图 5 所示实施例介绍的 FEC 编码。交织处理单元 305 用于执行上述图 18 所示实施例介绍的第三 FEC 交织处理。符号映射单元 306 和偏振划分单元 307 用于执行上述图 18 所示实施例介绍的符号映射和偏振划分。

应理解，本申请提供的装置也可以通过其他方式实现。例如，上述装置中的单元划分仅仅是一种逻辑功能划分，实际实现时可以有另外的划分方式，例如，多个单元或组件可以结合或可以集成到另一个系统。另外，本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个独立的物理单元，也可以是两个或两个以上个功能单元集成在一个处理单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

图 28 为本申请实施例中数据处理装置的第四种结构示意图。如图 28 所示，数据处理装置包括处理器 401、存储器 402 和收发器 403。该处理器 401、存储器 402 和收发器 403 通过线路相互连接。其中，存储器 402 用于存储程序指令和数据。具体地，处理器 401 用于执行数据处理的操作，收发器 403 用于执行数据收发的操作。在一种可能的实施方式中，处理器 401 可以包括上述图 25 所示的 PCS 单元 101、FEC 编码单元 102 和第一交织处理单元 103。在另一种可能的实施方式中，处理器 401 可以包括上述图 26 所示的第一处理单元 201、第二处理单元 202、加扰单元 203、第三处理单元 204、PCS 单元 205 和 FEC 编码单元 206。在又一种可能的实施方式中，处理器 401 可以包括上述图 27 所示的第一 PCS 单元 301、第一 FEC 编码单元 302、第二 PCS 单元 303、第二 FEC 单元 304、交织处理单元 305、符号映射单元 306 和偏振划分单元 307。

需要说明的是，上述图 28 中所示的处理器可以采用通用的中央处理器（Central Processing Unit，CPU），通用处理器、数字信号处理器（DSP）、专用集成电路（ASIC）、现场可编程门阵列（FPGA）或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。上述图 28 中所示的存储器可以存储操作系统和其他应用程序。在通过软件或者固件来实现本申请实施例提供的技术方案时，用于实现本申请实施例提供的技术方案的程序代码保存在存储器中，并由处理器来执行。在一实施例中，处理器内部可以包括存储器。在另一实施例中，处理器和存储器是两个独立的结构。

所属领域的技术人员可以清楚地了解到，为描述的方便和简洁，上述描述的系统，装置和单元的具体工作过程，可以参考前述方法实施例中的对应过程，在此不再赘述。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成，也可以通过程序来指令相关的硬件完成，所述的程序可以存储于一种计算机可读存储介质中，上述提到的存储介质可以是只读存储器，随机接入存储器等。上述的这些功能究竟以硬件还是软件方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

当使用软件实现时，上述实施例描述的方法步骤可以全部或部分地以计算机程序产品的形式实现。所述计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行所述计算机程序指令时，全部或部分地产生按照本申请实施例所述的流程或功能。所述计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。所述计算机指令可以存储在计算机可读存储介质中，或者从一个计算机可读存储介质向另一个计算机可读存储介质传输，例如，所述计算机指令可以从一个网站站点、计算机、服务器或数据中心通过有线（例如同轴电缆、光纤、数字用户线（DSL））或无线（例如红外、无线、微波等）方式向另一个网站站点、计算机、服务器或数据中心进行传输。所述计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。所述可用介质可以是磁性介质，（例如，软盘、硬盘、磁带）、光介质（例如，DVD）、或者半导体介质（例如固态硬盘 Solid State Disk（SSD））等。

权 利 要 求 书

1. 一种数据处理方法，其特征在于，包括：
 - 对 k 个比特中的第一比特集合进行概率星座整形 PCS 处理得到第二比特集合， k 为大于 1 的整数；
 - 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行前向纠错 FEC 编码得到第四比特集合，所述第四比特集合包括 m_0 个第一比特子集，每个所述第一比特子集包括 F_0 个比特，所述 m_0 为大于 1 的整数，所述 F_0 为大于 1 的偶数；
 - 对所述第四比特集合进行第一交织处理得到第五比特集合，所述第五比特集合包括 m_0 个第二比特子集，每个所述第二比特子集包括 F_0 个比特，每个所述第二比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合，每个所述第二比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。
2. 根据权利要求 1 所述的方法，其特征在于，所述第五比特集合包括 $m_0 \times F_0$ 个比特，每个所述第二比特子集包括连续的 F_0 个比特。
3. 根据权利要求 1 所述的方法，其特征在于，每个所述第二比特子集呈现为方阵。
4. 根据权利要求 1 至 3 中任一项所述的方法，其特征在于，每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列，所述第二比特子集的每行 F_1 个比特中的其中 $F_1/2$ 个比特来自所述第二比特集合，所述第二比特集合包括 $m_0 \times F_0/2$ 个比特，所述第二比特子集的每行 F_1 个比特中的另外 $F_1/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。
5. 根据权利要求 4 所述的方法，其特征在于， $F_0 = 256$ ， $F_1 = 16$ 。
6. 根据权利要求 4 或 5 所述的方法，其特征在于， $F_0 = 256$ ， $F_1 = 16$ ，所述第二比特子集中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特来自所述第二比特集合， $0 \leq i_2 < 16$ ， $0 \leq j_2 < 8$ ；
- 或者，所述第二比特子集中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自所述第二比特集合， $0 \leq i_2 < 16$ ， $0 \leq j_2 < 8$ ；
- 其中， $\lfloor a \rfloor$ 表示对 a 进行向下取整， $b \% c$ 表示 b 对 c 取模。
7. 根据权利要求 4 或 5 所述的方法，其特征在于， $F_0 = 256$ ， $F_1 = 16$ ，所述第二比特子集中的比特满足以下的一项或多项：
 - 所述第二比特子集中第 0 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 1 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 2 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 3 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 4 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 5 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 6 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 7 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 8 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 9 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 10 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 11 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 12 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 13 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 14 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合；
 - 所述第二比特子集中第 15 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合。
8. 根据权利要求 1 至 7 中任一项所述的方法，其特征在于， $F_0 = 256$ ，所述第四比特集合中的 m_1 个所述第一比特子集总共 $m_1 \times 256$ 个比特来自所述第二比特集合， $m_0 = m_1 \times 2$ 。
9. 根据权利要求 1 至 8 中任一项所述的方法，其特征在于， $F_0 = 256$ ，所述第四比特集合中 $m_0 \times 256$

个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中, $0 \leq r_0 < 32$, $0 \leq r_1 < 32$ 。

10. 根据权利要求 1 至 9 中任一项所述的方法, 其特征在于, 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合包括:

对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数;

对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

11. 根据权利要求 10 所述的方法, 其特征在于, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 5 列的每个第三比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第三比特子集中每行的另外 7 个比特来自所述第三比特集合, 第 6 列的每个第三比特子集中的 F_2 个比特来自所述第二比特集合。

12. 根据权利要求 9 至 11 中任一项所述的方法, 其特征在于, $r_0 = r_1$ 。

13. 根据权利要求 1 至 12 中任一项所述的方法, 其特征在于, $m_0=12$ 或 16。

14. 根据权利要求 1 至 13 中任一项所述的方法, 其特征在于, 对所述第四比特集合进行交织处理得到第五比特集合之后, 所述方法还包括:

对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数;

对 $L/2$ 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数;

对所述双偏振符号流进行数字信号处理 DSP 成帧。

15. 根据权利要求 14 所述的方法, 其特征在于, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

16. 根据权利要求 15 所述的方法, 其特征在于, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

17. 根据权利要求 16 所述的方法, 其特征在于, 对所述双偏振符号流进行 DSP 成帧包括:

将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

18. 根据权利要求 1 至 17 中任一项所述的方法, 其特征在于, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

19. 一种数据处理方法, 其特征在于, 包括:

从数据帧中获取第一数据, 所述第一数据包括 r 行 q 列个比特, 所述 r 为大于 0 的整数, 所述 q 为大于 0 的整数;

对所述第一数据进行循环冗余校验 CRC 和/或插入填充比特得到第二数据, 所述第二数据包括进行 CRC 插入的 d_{CRC} 个校验比特和/或 d_{PAD} 个填充比特, d_{CRC} 为大于或等于 0 的整数, d_{PAD} 为大于或等于 0 的整数;

对所述第二数据进行加扰得到第三数据, 所述第三数据的比特数量是 $d_{scr} = r \times q + d_{CP}$, $d_{CP} = d_{CRC} + d_{PAD}$;

获取所述第三数据中的 k 个比特, k 为大于 1 的整数, d_{scr} 是 k 的整数倍;

对 k 个比特中的第一比特集合进行概率星座整形 PCS 处理得到第二比特集合;

对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行前向纠错 FEC 编

码得到第四比特集合。

20. 根据权利要求 19 所述的方法, 其特征在于, 所述第四比特集合包括 m_0 个第一比特子集, 每个所述第一比特子集包括 F_0 个比特, 所述 m_0 为大于 1 的整数, 所述 F_0 为大于 1 的偶数, 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合包括:

5 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数;

10 对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

21. 根据权利要求 19 或 20 所述的方法, 其特征在于, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

22. 根据权利要求 19 至 21 中任一项所述的方法, 其特征在于, d_{scr} 是 $4 \times k$ 的整数倍, $q=10280$ 。

15 23. 根据权利要求 19 至 22 中任一项所述的方法, 其特征在于, $r=79$, $d_{CP}=328$, $k=3224$;

或者, $r=83$, $d_{CP}=32$, $k=3386$;

或者, $r=83$, $d_{CP}=536$, $k=3388$;

或者, $r=87$, $d_{CP}=72$, $k=2662$;

或者, $r=87$, $d_{CP}=744$, $k=2664$;

20 或者, $r=87$, $d_{CP}=3432$, $k=2672$;

或者, $r=104$, $d_{CP}=704$, $k=3184$;

或者, $r=104$, $d_{CP}=3392$, $k=3192$;

或者, $r=105$, $d_{CP}=504$, $k=3214$;

或者, $r=105$, $d_{CP}=1176$, $k=3216$;

25 或者, $r=109$, $d_{CP}=40$, $k=2668$;

或者, $r=109$, $d_{CP}=880$, $k=2670$;

或者, $r=110$, $d_{CP}=176$, $k=3366$;

或者, $r=110$, $d_{CP}=848$, $k=3368$;

或者, $r=110$, $d_{CP}=3536$, $k=3376$;

30 或者, $r=111$, $d_{CP}=648$, $k=3398$;

或者, $r=131$, $d_{CP}=8$, $k=2672$;

或者, $r=131$, $d_{CP}=1016$, $k=2674$;

或者, $r=131$, $d_{CP}=680$, $k=3208$;

或者, $r=132$, $d_{CP}=480$, $k=3232$;

35 或者, $r=132$, $d_{CP}=1320$, $k=3234$;

或者, $r=137$, $d_{CP}=320$, $k=3354$;

或者, $r=137$, $d_{CP}=1160$, $k=3356$;

或者, $r=138$, $d_{CP}=120$, $k=3378$;

或者, $r=138$, $d_{CP}=960$, $k=3380$;

40 或者, $r=152$, $d_{CP}=344$, $k=2658$;

或者, $r=152$, $d_{CP}=1520$, $k=2660$;

或者, $r=153$, $d_{CP}=648$, $k=2676$;

或者, $r=157$, $d_{CP}=856$, $k=3204$;

或者, $r=158$, $d_{CP}=656$, $k=3224$;

45 或者, $r=165$, $d_{CP}=264$, $k=3366$;

或者, $r=165$, $d_{CP}=1272$, $k=3368$;

- 或者, $r=166$, $d_{CP}=64$, $k=3386$;
或者, $r=166$, $d_{CP}=1072$, $k=3388$;
或者, $r=174$, $d_{CP}=144$, $k=2662$;
或者, $r=174$, $d_{CP}=1488$, $k=2664$;
5 或者, $r=175$, $d_{CP}=616$, $k=2678$;
或者, $r=183$, $d_{CP}=360$, $k=3200$;
或者, $r=183$, $d_{CP}=1536$, $k=3202$;
或者, $r=184$, $d_{CP}=664$, $k=3218$;
或者, $r=184$, $d_{CP}=1840$, $k=3220$;
10 或者, $r=185$, $d_{CP}=968$, $k=3236$;
或者, $r=192$, $d_{CP}=744$, $k=3358$;
或者, $r=192$, $d_{CP}=1920$, $k=3360$;
或者, $r=193$, $d_{CP}=1048$, $k=3376$;
或者, $r=194$, $d_{CP}=176$, $k=3392$;
15 或者, $r=194$, $d_{CP}=1352$, $k=3394$;
或者, $r=196$, $d_{CP}=616$, $k=2666$;
或者, $r=209$, $d_{CP}=536$, $k=3198$;
或者, $r=209$, $d_{CP}=1880$, $k=3200$;
或者, $r=210$, $d_{CP}=1008$, $k=3214$;
20 或者, $r=211$, $d_{CP}=136$, $k=3228$;
或者, $r=211$, $d_{CP}=1480$, $k=3230$;
或者, $r=217$, $d_{CP}=280$, $k=2656$;
或者, $r=217$, $d_{CP}=1960$, $k=2658$;
或者, $r=218$, $d_{CP}=80$, $k=2668$;
25 或者, $r=218$, $d_{CP}=1760$, $k=2670$;
或者, $r=219$, $d_{CP}=1224$, $k=3352$;
或者, $r=220$, $d_{CP}=352$, $k=3366$;
或者, $r=220$, $d_{CP}=1696$, $k=3368$;
或者, $r=221$, $d_{CP}=824$, $k=3382$;
30 或者, $r=221$, $d_{CP}=2168$, $k=3384$;
或者, $r=222$, $d_{CP}=1296$, $k=3398$;
或者, $r=235$, $d_{CP}=376$, $k=3196$;
或者, $r=235$, $d_{CP}=1888$, $k=3198$;
或者, $r=236$, $d_{CP}=680$, $k=3210$;
35 或者, $r=236$, $d_{CP}=2192$, $k=3212$;
或者, $r=237$, $d_{CP}=984$, $k=3224$;
或者, $r=238$, $d_{CP}=1288$, $k=3238$;
或者, $r=239$, $d_{CP}=920$, $k=2660$;
或者, $r=240$, $d_{CP}=1728$, $k=2672$;
40 或者, $r=247$, $d_{CP}=1000$, $k=3360$;
或者, $r=247$, $d_{CP}=2512$, $k=3362$;
或者, $r=248$, $d_{CP}=1304$, $k=3374$;
或者, $r=249$, $d_{CP}=96$, $k=3386$;
或者, $r=249$, $d_{CP}=1608$, $k=3388$;
45 或者, $r=250$, $d_{CP}=400$, $k=3400$;
或者, $r=250$, $d_{CP}=1912$, $k=3402$;

- 或者, $r=261$, $d_{CP}=216$, $k=2662$;
或者, $r=261$, $d_{CP}=2232$, $k=2664$;
或者, $r=261$, $d_{CP}=1560$, $k=3196$;
或者, $r=262$, $d_{CP}=16$, $k=2672$;
5 或者, $r=262$, $d_{CP}=2032$, $k=2674$;
或者, $r=262$, $d_{CP}=1360$, $k=3208$;
或者, $r=263$, $d_{CP}=1160$, $k=3220$;
或者, $r=264$, $d_{CP}=960$, $k=3232$;
或者, $r=264$, $d_{CP}=2640$, $k=3234$;
10 或者, $r=274$, $d_{CP}=640$, $k=3354$;
或者, $r=274$, $d_{CP}=2320$, $k=3356$;
或者, $r=275$, $d_{CP}=440$, $k=3366$;
或者, $r=275$, $d_{CP}=2120$, $k=3368$;
或者, $r=276$, $d_{CP}=240$, $k=3378$;
15 或者, $r=276$, $d_{CP}=1920$, $k=3380$;
或者, $r=277$, $d_{CP}=40$, $k=3390$;
或者, $r=277$, $d_{CP}=1720$, $k=3392$;
或者, $r=283$, $d_{CP}=2032$, $k=2666$;
或者, $r=284$, $d_{CP}=488$, $k=2674$;
20 或者, $r=284$, $d_{CP}=2672$, $k=2676$;
或者, $r=287$, $d_{CP}=896$, $k=3194$;
或者, $r=287$, $d_{CP}=2744$, $k=3196$;
或者, $r=288$, $d_{CP}=1704$, $k=3206$;
或者, $r=289$, $d_{CP}=664$, $k=3216$;
25 或者, $r=289$, $d_{CP}=2512$, $k=3218$;
或者, $r=290$, $d_{CP}=1472$, $k=3228$;
或者, $r=291$, $d_{CP}=432$, $k=3238$;
或者, $r=291$, $d_{CP}=2280$, $k=3240$;
或者, $r=302$, $d_{CP}=80$, $k=3360$;
30 或者, $r=302$, $d_{CP}=1928$, $k=3362$;
或者, $r=303$, $d_{CP}=888$, $k=3372$;
或者, $r=303$, $d_{CP}=2736$, $k=3374$;
或者, $r=304$, $d_{CP}=688$, $k=2658$;
或者, $r=304$, $d_{CP}=3040$, $k=2660$;
35 或者, $r=304$, $d_{CP}=1696$, $k=3384$;
或者, $r=305$, $d_{CP}=2168$, $k=2668$;
或者, $r=305$, $d_{CP}=656$, $k=3394$;
或者, $r=305$, $d_{CP}=2504$, $k=3396$;
或者, $r=306$, $d_{CP}=1296$, $k=2676$;
40 或者, $r=313$, $d_{CP}=1912$, $k=3194$;
或者, $r=314$, $d_{CP}=1712$, $k=3204$;
或者, $r=315$, $d_{CP}=1512$, $k=3214$;
或者, $r=316$, $d_{CP}=1312$, $k=3224$;
或者, $r=317$, $d_{CP}=1112$, $k=3234$;
45 或者, $r=317$, $d_{CP}=3128$, $k=3236$;
或者, $r=326$, $d_{CP}=320$, $k=2660$;

- 或者, $r=326$, $d_{CP}=2840$, $k=2662$;
或者, $r=327$, $d_{CP}=120$, $k=2668$;
或者, $r=327$, $d_{CP}=2640$, $k=2670$;
或者, $r=328$, $d_{CP}=2440$, $k=2678$;
5 或者, $r=329$, $d_{CP}=728$, $k=3356$;
或者, $r=329$, $d_{CP}=2744$, $k=3358$;
或者, $r=330$, $d_{CP}=528$, $k=3366$;
或者, $r=330$, $d_{CP}=2544$, $k=3368$;
或者, $r=331$, $d_{CP}=328$, $k=3376$;
10 或者, $r=331$, $d_{CP}=2344$, $k=3378$;
或者, $r=332$, $d_{CP}=128$, $k=3386$;
或者, $r=332$, $d_{CP}=2144$, $k=3388$;
或者, $r=333$, $d_{CP}=1944$, $k=3398$;
或者, $r=339$, $d_{CP}=744$, $k=3192$;
15 或者, $r=339$, $d_{CP}=2928$, $k=3194$;
或者, $r=340$, $d_{CP}=1384$, $k=3202$;
或者, $r=341$, $d_{CP}=2024$, $k=3212$;
或者, $r=342$, $d_{CP}=480$, $k=3220$;
或者, $r=342$, $d_{CP}=2664$, $k=3222$;
20 或者, $r=343$, $d_{CP}=1120$, $k=3230$;
或者, $r=343$, $d_{CP}=3304$, $k=3232$;
或者, $r=344$, $d_{CP}=1760$, $k=3240$;
或者, $r=348$, $d_{CP}=288$, $k=2662$;
或者, $r=348$, $d_{CP}=2976$, $k=2664$;
25 或者, $r=349$, $d_{CP}=760$, $k=2670$;
或者, $r=349$, $d_{CP}=3448$, $k=2672$;
或者, $r=350$, $d_{CP}=1232$, $k=2678$;
或者, $r=356$, $d_{CP}=704$, $k=3352$;
或者, $r=356$, $d_{CP}=2888$, $k=3354$;
30 或者, $r=357$, $d_{CP}=1344$, $k=3362$;
或者, $r=357$, $d_{CP}=3528$, $k=3364$;
或者, $r=358$, $d_{CP}=1984$, $k=3372$;
或者, $r=359$, $d_{CP}=440$, $k=3380$;
或者, $r=359$, $d_{CP}=2624$, $k=3382$;
35 或者, $r=360$, $d_{CP}=1080$, $k=3390$;
或者, $r=360$, $d_{CP}=3264$, $k=3392$;
或者, $r=361$, $d_{CP}=1720$, $k=3400$;
或者, $r=365$, $d_{CP}=1592$, $k=3192$;
或者, $r=366$, $d_{CP}=720$, $k=3200$;
40 或者, $r=366$, $d_{CP}=3072$, $k=3202$;
或者, $r=367$, $d_{CP}=2200$, $k=3210$;
或者, $r=368$, $d_{CP}=1328$, $k=3218$;
或者, $r=368$, $d_{CP}=3680$, $k=3220$;
或者, $r=369$, $d_{CP}=2304$, $k=2658$;
45 或者, $r=369$, $d_{CP}=456$, $k=3226$;
或者, $r=369$, $d_{CP}=2808$, $k=3228$;

- 或者, $r=370$, $d_{CP}=592$, $k=2664$;
或者, $r=370$, $d_{CP}=3448$, $k=2666$;
或者, $r=370$, $d_{CP}=1936$, $k=3236$;
或者, $r=371$, $d_{CP}=1736$, $k=2672$;
5 或者, $r=384$, $d_{CP}=1488$, $k=3358$;
或者, $r=384$, $d_{CP}=3840$, $k=3360$;
或者, $r=385$, $d_{CP}=616$, $k=3366$;
或者, $r=385$, $d_{CP}=2968$, $k=3368$;
或者, $r=386$, $d_{CP}=2096$, $k=3376$;
10 或者, $r=387$, $d_{CP}=1224$, $k=3384$;
或者, $r=387$, $d_{CP}=3576$, $k=3386$;
或者, $r=388$, $d_{CP}=352$, $k=3392$;
或者, $r=388$, $d_{CP}=2704$, $k=3394$;
或者, $r=389$, $d_{CP}=1832$, $k=3402$;
15 或者, $r=391$, $d_{CP}=2440$, $k=2660$;
或者, $r=392$, $d_{CP}=1232$, $k=2666$;
或者, $r=392$, $d_{CP}=2240$, $k=3200$;
或者, $r=393$, $d_{CP}=24$, $k=2672$;
或者, $r=393$, $d_{CP}=3048$, $k=2674$;
20 或者, $r=393$, $d_{CP}=2040$, $k=3208$;
或者, $r=394$, $d_{CP}=1840$, $k=3216$;
或者, $r=395$, $d_{CP}=1640$, $k=3224$;
或者, $r=396$, $d_{CP}=1440$, $k=3232$;
或者, $r=396$, $d_{CP}=3960$, $k=3234$;
25 或者, $r=397$, $d_{CP}=1240$, $k=3240$;
或者, $r=397$, $d_{CP}=3760$, $k=3242$;
或者, $r=411$, $d_{CP}=960$, $k=3354$;
或者, $r=411$, $d_{CP}=3480$, $k=3356$;
或者, $r=412$, $d_{CP}=760$, $k=3362$;
30 或者, $r=412$, $d_{CP}=3280$, $k=3364$;
或者, $r=413$, $d_{CP}=2912$, $k=2662$;
或者, $r=413$, $d_{CP}=560$, $k=3370$;
或者, $r=413$, $d_{CP}=3080$, $k=3372$;
或者, $r=414$, $d_{CP}=2208$, $k=2668$;
35 或者, $r=414$, $d_{CP}=360$, $k=3378$;
或者, $r=414$, $d_{CP}=2880$, $k=3380$;
或者, $r=415$, $d_{CP}=1504$, $k=2674$;
或者, $r=415$, $d_{CP}=160$, $k=3386$;
或者, $r=415$, $d_{CP}=2680$, $k=3388$;
40 或者, $r=416$, $d_{CP}=2480$, $k=3396$;
或者, $r=418$, $d_{CP}=1072$, $k=3198$;
或者, $r=418$, $d_{CP}=3760$, $k=3200$;
或者, $r=419$, $d_{CP}=1544$, $k=3206$;
或者, $r=419$, $d_{CP}=4232$, $k=3208$;
45 或者, $r=420$, $d_{CP}=2016$, $k=3214$;
或者, $r=421$, $d_{CP}=2488$, $k=3222$;

- 或者, $r=422$, $d_{CP}=272$, $k=3228$;
或者, $r=422$, $d_{CP}=2960$, $k=3230$;
或者, $r=423$, $d_{CP}=744$, $k=3236$;
或者, $r=423$, $d_{CP}=3432$, $k=3238$;
5 或者, $r=434$, $d_{CP}=560$, $k=2656$;
或者, $r=434$, $d_{CP}=3920$, $k=2658$;
或者, $r=435$, $d_{CP}=360$, $k=2662$;
或者, $r=435$, $d_{CP}=3720$, $k=2664$;
或者, $r=436$, $d_{CP}=160$, $k=2668$;
10 或者, $r=436$, $d_{CP}=3520$, $k=2670$;
或者, $r=437$, $d_{CP}=3320$, $k=2676$;
或者, $r=438$, $d_{CP}=2448$, $k=3352$;
或者, $r=439$, $d_{CP}=232$, $k=3358$;
或者, $r=439$, $d_{CP}=2920$, $k=3360$;
15 或者, $r=440$, $d_{CP}=704$, $k=3366$;
或者, $r=440$, $d_{CP}=3392$, $k=3368$;
或者, $r=441$, $d_{CP}=1176$, $k=3374$;
或者, $r=441$, $d_{CP}=3864$, $k=3376$;
或者, $r=442$, $d_{CP}=1648$, $k=3382$;
20 或者, $r=442$, $d_{CP}=4336$, $k=3384$;
或者, $r=443$, $d_{CP}=2120$, $k=3390$;
或者, $r=444$, $d_{CP}=2424$, $k=3198$;
或者, $r=444$, $d_{CP}=2592$, $k=3398$;
或者, $r=445$, $d_{CP}=712$, $k=3204$;
25 或者, $r=445$, $d_{CP}=3568$, $k=3206$;
或者, $r=446$, $d_{CP}=1856$, $k=3212$;
或者, $r=447$, $d_{CP}=144$, $k=3218$;
或者, $r=447$, $d_{CP}=3000$, $k=3220$;
或者, $r=448$, $d_{CP}=1288$, $k=3226$;
30 或者, $r=448$, $d_{CP}=4144$, $k=3228$;
或者, $r=449$, $d_{CP}=2432$, $k=3234$;
或者, $r=456$, $d_{CP}=1032$, $k=2658$;
或者, $r=456$, $d_{CP}=4560$, $k=2660$;
或者, $r=457$, $d_{CP}=1336$, $k=2664$;
35 或者, $r=458$, $d_{CP}=1640$, $k=2670$;
或者, $r=459$, $d_{CP}=1944$, $k=2676$;
或者, $r=466$, $d_{CP}=1888$, $k=3356$;
或者, $r=466$, $d_{CP}=4744$, $k=3358$;
或者, $r=467$, $d_{CP}=176$, $k=3362$;
40 或者, $r=467$, $d_{CP}=3032$, $k=3364$;
或者, $r=468$, $d_{CP}=1320$, $k=3370$;
或者, $r=468$, $d_{CP}=4176$, $k=3372$;
或者, $r=469$, $d_{CP}=2464$, $k=3378$;
或者, $r=470$, $d_{CP}=752$, $k=3196$;
45 或者, $r=470$, $d_{CP}=3776$, $k=3198$;
或者, $r=470$, $d_{CP}=752$, $k=3384$;

- 或者, $r=470$, $d_{CP}=3608$, $k=3386$;
或者, $r=471$, $d_{CP}=2568$, $k=3204$;
或者, $r=471$, $d_{CP}=1896$, $k=3392$;
或者, $r=471$, $d_{CP}=4752$, $k=3394$;
5 或者, $r=472$, $d_{CP}=1360$, $k=3210$;
或者, $r=472$, $d_{CP}=4384$, $k=3212$;
或者, $r=472$, $d_{CP}=184$, $k=3398$;
或者, $r=472$, $d_{CP}=3040$, $k=3400$;
或者, $r=473$, $d_{CP}=152$, $k=3216$;
10 或者, $r=473$, $d_{CP}=3176$, $k=3218$;
或者, $r=474$, $d_{CP}=1968$, $k=3224$;
或者, $r=475$, $d_{CP}=760$, $k=3230$;
或者, $r=475$, $d_{CP}=3784$, $k=3232$;
或者, $r=476$, $d_{CP}=2576$, $k=3238$;
15 或者, $r=478$, $d_{CP}=1840$, $k=2660$;
或者, $r=479$, $d_{CP}=2648$, $k=2666$;
或者, $r=480$, $d_{CP}=3456$, $k=2672$;
或者, $r=481$, $d_{CP}=568$, $k=2676$;
或者, $r=481$, $d_{CP}=4264$, $k=2678$;
20 或者, $r=493$, $d_{CP}=184$, $k=3352$;
或者, $r=493$, $d_{CP}=3208$, $k=3354$;
或者, $r=494$, $d_{CP}=2000$, $k=3360$;
或者, $r=494$, $d_{CP}=5024$, $k=3362$;
或者, $r=495$, $d_{CP}=792$, $k=3366$;
25 或者, $r=495$, $d_{CP}=3816$, $k=3368$;
或者, $r=496$, $d_{CP}=1936$, $k=3196$;
或者, $r=496$, $d_{CP}=2608$, $k=3374$;
或者, $r=497$, $d_{CP}=1232$, $k=3202$;
或者, $r=497$, $d_{CP}=4424$, $k=3204$;
30 或者, $r=497$, $d_{CP}=1400$, $k=3380$;
或者, $r=497$, $d_{CP}=4424$, $k=3382$;
或者, $r=498$, $d_{CP}=528$, $k=3208$;
或者, $r=498$, $d_{CP}=3720$, $k=3210$;
或者, $r=498$, $d_{CP}=192$, $k=3386$;
35 或者, $r=498$, $d_{CP}=3216$, $k=3388$;
或者, $r=499$, $d_{CP}=1672$, $k=2656$;
或者, $r=499$, $d_{CP}=3016$, $k=3216$;
或者, $r=499$, $d_{CP}=2008$, $k=3394$;
或者, $r=499$, $d_{CP}=5032$, $k=3396$;
40 或者, $r=500$, $d_{CP}=2984$, $k=2662$;
或者, $r=500$, $d_{CP}=2312$, $k=3222$;
或者, $r=500$, $d_{CP}=800$, $k=3400$;
或者, $r=500$, $d_{CP}=3824$, $k=3402$;
或者, $r=501$, $d_{CP}=432$, $k=2666$;
45 或者, $r=501$, $d_{CP}=4296$, $k=2668$;
或者, $r=501$, $d_{CP}=1608$, $k=3228$;

或者, $r=501$, $d_{CP}=4800$, $k=3230$;

或者, $r=502$, $d_{CP}=1744$, $k=2672$;

或者, $r=502$, $d_{CP}=904$, $k=3234$;

或者, $r=502$, $d_{CP}=4096$, $k=3236$;

5 或者, $r=503$, $d_{CP}=3056$, $k=2678$ 。

24. 根据权利要求 19 至 23 中任一项所述的方法, 其特征在于, $d_{scr} = 336 \times k$ 。

25. 根据权利要求 19 至 24 中任一项所述的方法, 其特征在于, 对所述第一数据进行 CRC 包括:

对所述第一数据中每 r/p 行总共 $r \times q/p$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特,

$d_{CRC} = 32 \times p$, r 能被 p 整除, p 为大于 1 的整数;

10 或者,

对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数 r_{F0} 大于整数 r_{F1} 。

15 26. 根据权利要求 19 至 25 中任一项所述的方法, 其特征在于, $q=2056$ 。

27. 根据权利要求 26 所述的方法, 其特征在于, 对所述第一数据进行 CRC 包括:

对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数

20 r_{F0} 大于整数 r_{F1} 。

28. 根据权利要求 26 或 27 所述的方法, 其特征在于, $r=435$ 。

29. 根据权利要求 28 所述的方法, 其特征在于, $p=22$, $r_{F0}=20$, $r_{F1}=15$, 对所述第一数据进行 CRC 包括:

对所述第一数据中前 420 行中的每 20 行共 41120 个比特进行 CRC-32 校验以添加长度为 32 比特的
25 校验比特, 并对所述第一数据中最后 15 行共 30840 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。

30. 根据权利要求 28 或 29 所述的方法, 其特征在于, $d_{CP} = 3432$, $k = 2672$, 或者, $d_{CP} = 744$, $k = 2664$ 。

31. 根据权利要求 26 或 27 所述的方法, 其特征在于, $r=546$, $d_{CP}=1008$, $k=3344$;

30 或者, $r=546$, $d_{CP}=3696$, $k=3352$;

或者, $r=547$, $d_{CP}=1640$, $k=3352$;

或者, $r=547$, $d_{CP}=4328$, $k=3360$;

或者, $r=548$, $d_{CP}=2272$, $k=3360$;

或者, $r=548$, $d_{CP}=4960$, $k=3368$;

35 或者, $r=549$, $d_{CP}=216$, $k=3360$;

或者, $r=549$, $d_{CP}=2904$, $k=3368$;

或者, $r=549$, $d_{CP}=5592$, $k=3376$;

或者, $r=550$, $d_{CP}=848$, $k=3368$;

或者, $r=550$, $d_{CP}=3536$, $k=3376$;

40 或者, $r=551$, $d_{CP}=1480$, $k=3376$;

或者, $r=551$, $d_{CP}=4168$, $k=3384$;

或者, $r=552$, $d_{CP}=2112$, $k=3384$;

或者, $r=552$, $d_{CP}=4800$, $k=3392$;

或者, $r=553$, $d_{CP}=56$, $k=3384$;

45 或者, $r=553$, $d_{CP}=2744$, $k=3392$;

或者, $r=553$, $d_{CP}=5432$, $k=3400$;

- 或者, $r=554$, $d_{CP}=688$, $k=3392$;
或者, $r=554$, $d_{CP}=3376$, $k=3400$;
或者, $r=555$, $d_{CP}=1320$, $k=3400$;
或者, $r=555$, $d_{CP}=4008$, $k=3408$;
5 或者, $r=556$, $d_{CP}=1952$, $k=3408$;
或者, $r=556$, $d_{CP}=4640$, $k=3416$;
或者, $r=557$, $d_{CP}=2584$, $k=3416$;
或者, $r=557$, $d_{CP}=5272$, $k=3424$;
或者, $r=558$, $d_{CP}=528$, $k=3416$;
10 或者, $r=558$, $d_{CP}=3216$, $k=3424$;
或者, $r=559$, $d_{CP}=1160$, $k=3424$;
或者, $r=559$, $d_{CP}=3848$, $k=3432$;
或者, $r=560$, $d_{CP}=1792$, $k=3432$;
或者, $r=560$, $d_{CP}=4480$, $k=3440$;
15 或者, $r=561$, $d_{CP}=2424$, $k=3440$;
或者, $r=561$, $d_{CP}=5112$, $k=3448$;
或者, $r=562$, $d_{CP}=368$, $k=3440$;
或者, $r=562$, $d_{CP}=3056$, $k=3448$;
或者, $r=562$, $d_{CP}=5744$, $k=3456$ 。
20 32. 根据权利要求 26 或 27 所述的方法, 其特征在于, $r=520$, $d_{CP}=704$, $k=3184$;
或者, $r=520$, $d_{CP}=3392$, $k=3192$;
或者, $r=520$, $d_{CP}=6080$, $k=3200$;
或者, $r=521$, $d_{CP}=1336$, $k=3192$;
或者, $r=521$, $d_{CP}=4024$, $k=3200$;
25 或者, $r=522$, $d_{CP}=1968$, $k=3200$;
或者, $r=522$, $d_{CP}=4656$, $k=3208$;
或者, $r=523$, $d_{CP}=2600$, $k=3208$;
或者, $r=523$, $d_{CP}=5288$, $k=3216$;
或者, $r=524$, $d_{CP}=544$, $k=3208$;
30 或者, $r=524$, $d_{CP}=3232$, $k=3216$;
或者, $r=525$, $d_{CP}=1176$, $k=3216$;
或者, $r=525$, $d_{CP}=3864$, $k=3224$;
或者, $r=526$, $d_{CP}=1808$, $k=3224$;
或者, $r=526$, $d_{CP}=4496$, $k=3232$;
35 或者, $r=527$, $d_{CP}=2440$, $k=3232$;
或者, $r=527$, $d_{CP}=5128$, $k=3240$;
或者, $r=528$, $d_{CP}=384$, $k=3232$;
或者, $r=528$, $d_{CP}=3072$, $k=3240$;
或者, $r=528$, $d_{CP}=5760$, $k=3248$;
40 或者, $r=529$, $d_{CP}=1016$, $k=3240$;
或者, $r=529$, $d_{CP}=3704$, $k=3248$;
或者, $r=530$, $d_{CP}=1648$, $k=3248$;
或者, $r=530$, $d_{CP}=4336$, $k=3256$;
或者, $r=531$, $d_{CP}=2280$, $k=3256$;
45 或者, $r=531$, $d_{CP}=4968$, $k=3264$;
或者, $r=532$, $d_{CP}=2912$, $k=3264$;

或者, $r=532$, $d_{CP}=224$, $k=3256$;
 或者, $r=533$, $d_{CP}=856$, $k=3264$;
 或者, $r=533$, $d_{CP}=3544$, $k=3272$;
 或者, $r=534$, $d_{CP}=1488$, $k=3272$;
 5 或者, $r=534$, $d_{CP}=4176$, $k=3280$;
 或者, $r=535$, $d_{CP}=2120$, $k=3280$;
 或者, $r=535$, $d_{CP}=4808$, $k=3288$ 。

33. 根据权利要求 19 至 32 中任一项所述的方法, 其特征在于, 所述第四比特集合包括 m_0 个第一比特子集, 所述 m_0 为大于 1 的整数, 对所述第二比特集合和所述第三比特集合进行所述 FEC 编码得到第四比特集合之后, 所述方法还包括:

对所述第四比特集合进行第一交织处理得到第五比特集合, 所述第五比特集合包括 m_0 个第二比特子集, 每个所述第二比特子集包括 256 个比特, 每个所述第二比特子集中的其中 128 个比特来自所述第二比特集合, 每个所述第二比特子集中的另外 128 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

15 34. 根据权利要求 33 所述的方法, 其特征在于, 每个所述第二比特子集包括连续的 256 个比特。

35. 根据权利要求 33 所述的方法, 其特征在于, 每个所述第二比特子集呈现为方阵。

36. 根据权利要求 33 至 35 中任一项所述的方法, 其特征在于, 所述第二比特子集的 256 个比特分布为 16 行 16 列, 所述第二比特子集中的比特满足以下的一项或多项:

所述第二比特子集中第 0 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 20 所述第二比特子集中第 1 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 2 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 3 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
 所述第二比特子集中第 4 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 5 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 25 所述第二比特子集中第 6 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 7 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
 所述第二比特子集中第 8 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 9 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 10 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 30 所述第二比特子集中第 11 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
 所述第二比特子集中第 12 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 13 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 14 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 15 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合。

35 37. 根据权利要求 33 至 36 中任一项所述的方法, 其特征在于, 所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中, $0 \leq r_0 < 32$, $0 \leq r_1 < 32$ 。

38. 根据权利要求 37 所述的方法, 其特征在于, $r_0 = r_1$ 。

40 39. 根据权利要求 19 至 38 中任一项所述的方法, 其特征在于, 所述第四比特集合包括 m_0 个第一比特子集, $m_0=12$ 或 16。

40. 根据权利要求 33 至 39 中任一项所述的方法, 其特征在于, 对所述第四比特集合进行第一交织处理得到第五比特集合之后, 所述方法还包括:

对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数;

对 $L/2$ 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和

所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数;

对所述双偏振符号流进行数字信号处理 DSP 成帧。

41. 根据权利要求 40 所述的方法, 其特征在于, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

5 42. 根据权利要求 41 所述的方法, 其特征在于, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

43. 根据权利要求 42 所述的方法, 其特征在于, 对所述双偏振符号流进行 DSP 成帧包括:

将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

44. 根据权利要求 43 所述的方法, 其特征在于, 从数据帧中获取第一数据包括:

10 从所述数据帧中获取 F_{Frame} 个第一数据, 以得到 $F_{\text{Frame}} \times r \times q$ 个比特, F_{Frame} 为大于 0 的整数;

$$d_{\text{scr}} \times F_{\text{Frame}} = 336 \times k \times F_{\text{DSP}};$$

其中, F_{DSP} 表示所述 DSP 超帧的数量, F_{DSP} 个 DSP 超帧由 F_{Frame} 个第一数据经过处理得到, F_{DSP} 为大于 0 的整数。

45. 根据权利要求 44 所述的方法, 其特征在于, $F_{\text{DSP}} > 1$, 所述 F_{DSP} 个 DSP 超帧中第一 DSP 超帧中的连续多个符号为第一标识, 所述 F_{DSP} 个 DSP 超帧中除所述第一 DSP 超帧外的每个 DSP 超帧中的连续多个符号为第二标识。

46. 一种数据处理方法, 其特征在于, 包括:

对第一组 k 个比特中的第一比特集合进行第一概率星座整形 PCS 处理得到第二比特集合, 所述第二比特集合包括 $m_0 \times 128$ 个比特, k 为大于 1 的整数, 所述 m_0 为大于 1 的整数;

20 对第二比特集合和所述第一组 k 个比特中除所述第一比特集合外的第三比特集合进行第一前向纠错 FEC 编码得到第四比特集合, 所述第四比特集合包括 m_0 个第一比特方阵, 每个所述第一比特方阵包括分布为 16 行 16 列的共 256 个比特;

对第二组 k 个比特中的第五比特集合进行第二 PCS 处理得到第六比特集合;

25 对第六比特集合和所述第二组 k 个比特中除所述第五比特集合外的第七比特集合进行第二 FEC 编码得到第八比特集合, 所述第八比特集合包括 m_0 个第二比特方阵, 每个所述第二比特方阵包括分布为 16 行 16 列的共 256 个比特;

将 21 个所述第四比特集合和 21 个所述第八比特集合输入交织缓存, 其中, 所述交织缓存包括分布为 84 行 m_1 列的共 $84 \times m_1$ 个缓存单元, $m_0 = m_1 \times 2$, 每个所述缓存单元用于缓存 16 行 16 列共 256 个比特, $42 \times m_1$ 个所述第一比特方阵输入所述交织缓存的偶数行, $42 \times m_1$ 个所述第二比特方阵输入所述交织缓存的奇数行, 所述交织缓存包括第一缓存子集和第二缓存子集, 所述第一缓存子集包括 $42 \times m_1$ 个所述缓存单元, 所述第二缓存子集包括 $42 \times m_1$ 个所述缓存单元, 输入所述第一缓存子集的比特来自所述第二比特集合和所述第六比特集合, 输入所述第二缓存子集的比特来自所述第三比特集合、所述第七比特集合、第一 FEC 编码的校验比特和第二 FEC 编码的校验比特;

35 对所述交织缓存中每 8 个比特进行符号映射和偏振划分得到 1 个双偏振 DP-16QAM 符号, 所述 DP-16QAM 符号在目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个幅度位比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 DP-16QAM 符号在所述目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个符号位比特来自所述第二缓存子集的第二缓存单元中的一列。

47. 根据权利要求 46 所述的方法, 其特征在于, 所述 8 个比特中的其中 4 个比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 8 个比特中的另外 4 个比特来自所述第二缓存子集的第二缓存单元中的一列。

48. 根据权利要求 46 或 47 所述的方法, 其特征在于, 所述 m_0 个第一比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第一比特方阵和第 1 行第 m_2 列的第一比特方阵来自所述第二比特集合, 所述 m_0 个第二比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第二比特方阵和第 1 行第 m_2 列的第二比特方阵来自所述第六比特集合, $0 \leq m_2 < m_1/2$ 。

45 49. 根据权利要求 48 所述的方法, 其特征在于, $m_0=16$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 2 列和第 3 列中的共 336 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 4 列、

第 5 列、第 6 列和第 7 列中的共 336 个缓存单元；

或者， $m_0=12$ ，所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列和第 2 列中的共 252 个缓存单元，所述第二缓存子集包括所述交织缓存中第 3 列、第 4 列和第 5 列中的共 252 个缓存单元。

50. 根据权利要求 46 或 47 所述的方法，其特征在于， $m_0=16, m_1=8$ ；

5 所述 m_0 个第一比特方阵分布为 2 行 m_1 列，第 0 行第 0 列的第一比特方阵、第 0 行第 1 列的第一比特方阵、第 0 行第 4 列的第一比特方阵、第 0 行第 5 列的第一比特方阵、第 1 行第 0 列的第一比特方阵、第 1 行第 1 列的第一比特方阵、第 1 行第 4 列的第一比特方阵和第 1 行第 5 列的第一比特方阵来自所述第二比特集合；

10 所述 m_0 个第二比特方阵分布为 2 行 m_1 列，第 0 行第 0 列的第二比特方阵、第 0 行第 1 列的第二比特方阵、第 0 行第 4 列的第二比特方阵、第 0 行第 5 列的第二比特方阵、第 1 行第 0 列的第二比特方阵、第 1 行第 1 列的第二比特方阵、第 1 行第 4 列的第二比特方阵和第 1 行第 5 列的第二比特方阵来自所述第六比特集合。

15 51. 根据权利要求 50 所述的方法，其特征在于，所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 4 列和第 5 列中的共 336 个缓存单元，所述第二缓存子集包括所述交织缓存中第 2 列、第 3 列、第 6 列和第 7 列中的共 336 个缓存单元。

52. 一种数据处理装置，其特征在于，包括：概率星座整形 PCS 单元、前向纠错 FEC 编码单元和第一交织处理单元；

所述 PCS 单元用于：对 k 个比特中的第一比特集合进行 PCS 处理得到第二比特集合， k 为大于 1 的整数；

20 所述 FEC 单元用于：对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合，所述第四比特集合包括 m_0 个第一比特子集，每个所述第一比特子集包括 F_0 个比特，所述 m_0 为大于 1 的整数，所述 F_0 为大于 1 的偶数；

25 所述第一交织处理单元用于：对所述第四比特集合进行第一交织处理得到第五比特集合，所述第五比特集合包括 m_0 个第二比特子集，每个所述第二比特子集包括 F_0 个比特，每个所述第二比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合，每个所述第二比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

53. 根据权利要求 52 所述的数据处理装置，其特征在于，所述第五比特集合包括 $m_0 \times F_0$ 个比特，每个所述第二比特子集包括连续的 F_0 个比特。

54. 根据权利要求 52 所述的数据处理装置，其特征在于，每个所述第二比特子集呈现为方阵。

30 55. 根据权利要求 52 至 54 中任一项所述的数据处理装置，其特征在于，每个所述第二比特子集中的 F_0 个比特分布为 F_1 行 F_1 列，所述第二比特子集的每行 F_1 个比特中的其中 $F_1/2$ 个比特来自所述第二比特集合，所述第二比特集合包括 $m_0 \times F_0/2$ 个比特，所述第二比特子集的每行 F_1 个比特中的另外 $F_1/2$ 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

56. 根据权利要求 55 所述的数据处理装置，其特征在于， $F_0 = 256, F_1 = 16$ 。

35 57. 根据权利要求 55 或 56 所述的数据处理装置，其特征在于， $F_0 = 256, F_1 = 16$ ，所述第二比特子集中第 i_2 行第 $\left(\left\lfloor \frac{i_2}{2} \right\rfloor \times 4 + j_2 \% 2 + i_2 + 2\right) \% 16$ 列的比特来自所述第二比特集合， $0 \leq i_2 < 16, 0 \leq j_2 < 8$ ；

或者，所述第二比特子集中第 i_2 行第 $(j_2 \times 2 - i_2 \% 2 + 1)$ 列的比特来自所述第二比特集合， $0 \leq i_2 < 16, 0 \leq j_2 < 8$ ；

其中， $\lfloor a \rfloor$ 表示对 a 进行向下取整， $b \% c$ 表示 b 对 c 取模。

40 58. 根据权利要求 55 或 56 所述的数据处理装置，其特征在于， $F_0 = 256, F_1 = 16$ ，所述第二比特子集中的比特满足以下的一项或多项：

所述第二比特子集中第 0 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合；

所述第二比特子集中第 1 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合；

所述第二比特子集中第 2 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合；

45 所述第二比特子集中第 3 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合；

所述第二比特子集中第 4 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 5 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 6 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 7 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
 5 所述第二比特子集中第 8 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 9 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 10 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 11 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
 所述第二比特子集中第 12 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
 10 所述第二比特子集中第 13 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
 所述第二比特子集中第 14 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
 所述第二比特子集中第 15 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合。

59. 根据权利要求 52 至 58 中任一项所述的数据处理装置, 其特征在于, $F_0 = 256$, 所述第四比特集合中的 m_1 个所述第一比特子集总共 $m_1 \times 256$ 个比特来自所述第二比特集合, $m_0 = m_1 \times 2$ 。

15 60. 根据权利要求 52 至 59 中任一项所述的数据处理装置, 其特征在于, $F_0 = 256$, 所述第四比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中, $0 \leq r_0 < 32$, $0 \leq r_1 < 32$ 。

20 61. 根据权利要求 52 至 60 中任一项所述的数据处理装置, 其特征在于, 所述数据处理装置还包括编码前交织单元;

所述编码前交织单元用于: 对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数;

所述 FEC 单元用于: 对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

30 62. 根据权利要求 61 所述的数据处理装置, 其特征在于, $m_3=14$, $m_4=7$, $F_0=256$, $F_2=240$, 第 0 列至第 5 列的每个第三比特子集包括 16 行 16 列个比特, 第 6 列的每个第三比特子集包括 16 行 15 列个比特, 第 0 列至第 4 列的每个第三比特子集中的其中 $F_0/2$ 个比特来自所述第二比特集合, 第 0 列至第 4 列的每个第三比特子集中的另外 $F_0/2$ 个比特来自所述第三比特集合, 第 5 列的每个第三比特子集中每行的其中 9 个比特来自所述第二比特集合, 第 5 列的每个第三比特子集中每行的另外 7 个比特来自所述第三比特集合, 第 6 列的每个第三比特子集中的 F_2 个比特来自所述第二比特集合。

63. 根据权利要求 60 至 62 中任一项所述的数据处理装置, 其特征在于, $r_0 = r_1$ 。

35 64. 根据权利要求 52 至 63 任一项所述的数据处理装置, 其特征在于, $m_0=12$ 或 16。

65. 根据权利要求 52 至 64 中任一项所述的数据处理装置, 其特征在于, 所述数据处理装置还包括第二交织处理单元、符号映射单元、偏振划分单元和数字信号处理 DSP 成帧单元;

所述第二交织处理单元用于: 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 $L/2$ 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数;

40 所述符号映射单元和所述偏振划分单元用于: 对 $L/2$ 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数;

所述 DSP 成帧单元用于: 对所述双偏振符号流进行 DSP 成帧。

45 66. 根据权利要求 65 所述的数据处理装置, 其特征在于, 所述双偏振符号中的幅度位比特来自所述第二比特集合。

67. 根据权利要求 66 所述的数据处理装置, 其特征在于, $t=8$, 所述双偏振符号为双偏振 DP-16QAM

符号。

68. 根据权利要求 67 所述的数据处理装置, 其特征在于, 所述 DSP 成帧单元具体用于: 将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

69. 根据权利要求 52 至 68 中任一项所述的数据处理装置, 其特征在于, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

70. 一种数据处理装置, 其特征在于, 包括: 第一处理单元、第二处理单元、加扰单元、第三处理单元、概率星座整形 PCS 单元和前向纠错 FEC 编码单元;

所述第一处理单元用于: 从数据帧中获取第一数据, 所述第一数据包括 r 行 q 列个比特, 所述 r 为大于 0 的整数, 所述 q 为大于 0 的整数;

所述第二处理单元用于: 对所述第一数据进行循环冗余校验 CRC 和/或插入填充比特得到第二数据, 所述第二数据包括进行 CRC 插入的 d_{CRC} 个校验比特和/或 d_{PAD} 个填充比特, d_{CRC} 为大于或等于 0 的整数, d_{PAD} 为大于或等于 0 的整数;

加扰单元用于: 对所述第二数据进行加扰得到第三数据, 所述第三数据的比特数量是 $d_{scr} = r \times q + d_{CP}$, $d_{CP} = d_{CRC} + d_{PAD}$;

所述第三处理单元用于: 获取所述第三数据中的 k 个比特, k 为大于 1 的整数, d_{scr} 是 k 的整数倍;

所述 PCS 单元用于: 对 k 个比特中的第一比特集合进行 PCS 处理得到第二比特集合;

所述 FEC 单元用于: 对所述第二比特集合和所述 k 个比特中除所述第一比特集合外的第三比特集合进行 FEC 编码得到第四比特集合。

71. 根据权利要求 70 所述的数据处理装置, 其特征在于, 所述第四比特集合包括 m_0 个第一比特子集, 每个所述第一比特子集包括 F_0 个比特, 所述 m_0 为大于 1 的整数, 所述 F_0 为大于 1 的偶数, 所述数据处理装置还包括编码前交织单元, 所述编码前交织单元用于:

对所述第二比特集合和所述第三比特集合进行编码前交织处理得到第六比特集合, 所述第六比特集合中的比特数等于所述第二比特集合中的比特数与所述第三比特集合中的比特数相加, 所述第六比特集合包括 m_3 个第三比特子集, 所述 m_3 个第三比特子集分布为 2 行 m_4 列, $m_3 = 2 \times m_4$, 所述 m_3 为大于 1 且小于 m_0 的整数, 其中一部分第三比特子集中每个第三比特子集包括 F_0 个比特, 其中另一部分第三比特子集中每个第三比特子集包括 F_2 个比特, 所述 F_2 为大于 1 且小于 F_0 的偶数;

所述 FEC 单元用于: 对所述第六比特集合进行 FEC 编码得到所述第四比特集合。

72. 根据权利要求 70 或 71 所述的数据处理装置, 其特征在于, 所述第一比特集合中的比特数量是 2、4、8、或 16 的整数倍, 所述第二比特集合包括 2048 个比特, 所述第三比特集合包括 1504 比特, 所述第四比特集合包括 4096 个比特。

73. 根据权利要求 70 至 72 中任一项所述的数据处理装置, 其特征在于, d_{scr} 是 $4 \times k$ 的整数倍, $q=10280$ 。

74. 根据权利要求 70 至 73 中任一项所述的数据处理装置, 其特征在于, $r=79$, $d_{CP}=328$, $k=3224$;

或者, $r=83$, $d_{CP}=32$, $k=3386$;

或者, $r=83$, $d_{CP}=536$, $k=3388$;

或者, $r=87$, $d_{CP}=72$, $k=2662$;

或者, $r=87$, $d_{CP}=744$, $k=2664$;

或者, $r=87$, $d_{CP}=3432$, $k=2672$;

或者, $r=104$, $d_{CP}=704$, $k=3184$;

或者, $r=104$, $d_{CP}=3392$, $k=3192$;

或者, $r=105$, $d_{CP}=504$, $k=3214$;

或者, $r=105$, $d_{CP}=1176$, $k=3216$;

或者, $r=109$, $d_{CP}=40$, $k=2668$;

或者, $r=109$, $d_{CP}=880$, $k=2670$;

或者, $r=110$, $d_{CP}=176$, $k=3366$;

- 或者, $r=110$, $d_{CP}=848$, $k=3368$;
或者, $r=110$, $d_{CP}=3536$, $k=3376$;
或者, $r=111$, $d_{CP}=648$, $k=3398$;
或者, $r=131$, $d_{CP}=8$, $k=2672$;
5 或者, $r=131$, $d_{CP}=1016$, $k=2674$;
或者, $r=131$, $d_{CP}=680$, $k=3208$;
或者, $r=132$, $d_{CP}=480$, $k=3232$;
或者, $r=132$, $d_{CP}=1320$, $k=3234$;
或者, $r=137$, $d_{CP}=320$, $k=3354$;
10 或者, $r=137$, $d_{CP}=1160$, $k=3356$;
或者, $r=138$, $d_{CP}=120$, $k=3378$;
或者, $r=138$, $d_{CP}=960$, $k=3380$;
或者, $r=152$, $d_{CP}=344$, $k=2658$;
或者, $r=152$, $d_{CP}=1520$, $k=2660$;
15 或者, $r=153$, $d_{CP}=648$, $k=2676$;
或者, $r=157$, $d_{CP}=856$, $k=3204$;
或者, $r=158$, $d_{CP}=656$, $k=3224$;
或者, $r=165$, $d_{CP}=264$, $k=3366$;
或者, $r=165$, $d_{CP}=1272$, $k=3368$;
20 或者, $r=166$, $d_{CP}=64$, $k=3386$;
或者, $r=166$, $d_{CP}=1072$, $k=3388$;
或者, $r=174$, $d_{CP}=144$, $k=2662$;
或者, $r=174$, $d_{CP}=1488$, $k=2664$;
或者, $r=175$, $d_{CP}=616$, $k=2678$;
25 或者, $r=183$, $d_{CP}=360$, $k=3200$;
或者, $r=183$, $d_{CP}=1536$, $k=3202$;
或者, $r=184$, $d_{CP}=664$, $k=3218$;
或者, $r=184$, $d_{CP}=1840$, $k=3220$;
或者, $r=185$, $d_{CP}=968$, $k=3236$;
30 或者, $r=192$, $d_{CP}=744$, $k=3358$;
或者, $r=192$, $d_{CP}=1920$, $k=3360$;
或者, $r=193$, $d_{CP}=1048$, $k=3376$;
或者, $r=194$, $d_{CP}=176$, $k=3392$;
或者, $r=194$, $d_{CP}=1352$, $k=3394$;
35 或者, $r=196$, $d_{CP}=616$, $k=2666$;
或者, $r=209$, $d_{CP}=536$, $k=3198$;
或者, $r=209$, $d_{CP}=1880$, $k=3200$;
或者, $r=210$, $d_{CP}=1008$, $k=3214$;
或者, $r=211$, $d_{CP}=136$, $k=3228$;
40 或者, $r=211$, $d_{CP}=1480$, $k=3230$;
或者, $r=217$, $d_{CP}=280$, $k=2656$;
或者, $r=217$, $d_{CP}=1960$, $k=2658$;
或者, $r=218$, $d_{CP}=80$, $k=2668$;
或者, $r=218$, $d_{CP}=1760$, $k=2670$;
45 或者, $r=219$, $d_{CP}=1224$, $k=3352$;
或者, $r=220$, $d_{CP}=352$, $k=3366$;

- 或者, $r=220$, $d_{CP}=1696$, $k=3368$;
或者, $r=221$, $d_{CP}=824$, $k=3382$;
或者, $r=221$, $d_{CP}=2168$, $k=3384$;
或者, $r=222$, $d_{CP}=1296$, $k=3398$;
5 或者, $r=235$, $d_{CP}=376$, $k=3196$;
或者, $r=235$, $d_{CP}=1888$, $k=3198$;
或者, $r=236$, $d_{CP}=680$, $k=3210$;
或者, $r=236$, $d_{CP}=2192$, $k=3212$;
或者, $r=237$, $d_{CP}=984$, $k=3224$;
10 或者, $r=238$, $d_{CP}=1288$, $k=3238$;
或者, $r=239$, $d_{CP}=920$, $k=2660$;
或者, $r=240$, $d_{CP}=1728$, $k=2672$;
或者, $r=247$, $d_{CP}=1000$, $k=3360$;
或者, $r=247$, $d_{CP}=2512$, $k=3362$;
15 或者, $r=248$, $d_{CP}=1304$, $k=3374$;
或者, $r=249$, $d_{CP}=96$, $k=3386$;
或者, $r=249$, $d_{CP}=1608$, $k=3388$;
或者, $r=250$, $d_{CP}=400$, $k=3400$;
或者, $r=250$, $d_{CP}=1912$, $k=3402$;
20 或者, $r=261$, $d_{CP}=216$, $k=2662$;
或者, $r=261$, $d_{CP}=2232$, $k=2664$;
或者, $r=261$, $d_{CP}=1560$, $k=3196$;
或者, $r=262$, $d_{CP}=16$, $k=2672$;
或者, $r=262$, $d_{CP}=2032$, $k=2674$;
25 或者, $r=262$, $d_{CP}=1360$, $k=3208$;
或者, $r=263$, $d_{CP}=1160$, $k=3220$;
或者, $r=264$, $d_{CP}=960$, $k=3232$;
或者, $r=264$, $d_{CP}=2640$, $k=3234$;
或者, $r=274$, $d_{CP}=640$, $k=3354$;
30 或者, $r=274$, $d_{CP}=2320$, $k=3356$;
或者, $r=275$, $d_{CP}=440$, $k=3366$;
或者, $r=275$, $d_{CP}=2120$, $k=3368$;
或者, $r=276$, $d_{CP}=240$, $k=3378$;
或者, $r=276$, $d_{CP}=1920$, $k=3380$;
35 或者, $r=277$, $d_{CP}=40$, $k=3390$;
或者, $r=277$, $d_{CP}=1720$, $k=3392$;
或者, $r=283$, $d_{CP}=2032$, $k=2666$;
或者, $r=284$, $d_{CP}=488$, $k=2674$;
或者, $r=284$, $d_{CP}=2672$, $k=2676$;
40 或者, $r=287$, $d_{CP}=896$, $k=3194$;
或者, $r=287$, $d_{CP}=2744$, $k=3196$;
或者, $r=288$, $d_{CP}=1704$, $k=3206$;
或者, $r=289$, $d_{CP}=664$, $k=3216$;
或者, $r=289$, $d_{CP}=2512$, $k=3218$;
45 或者, $r=290$, $d_{CP}=1472$, $k=3228$;
或者, $r=291$, $d_{CP}=432$, $k=3238$;

- 或者, $r=291$, $d_{CP}=2280$, $k=3240$;
或者, $r=302$, $d_{CP}=80$, $k=3360$;
或者, $r=302$, $d_{CP}=1928$, $k=3362$;
或者, $r=303$, $d_{CP}=888$, $k=3372$;
5 或者, $r=303$, $d_{CP}=2736$, $k=3374$;
或者, $r=304$, $d_{CP}=688$, $k=2658$;
或者, $r=304$, $d_{CP}=3040$, $k=2660$;
或者, $r=304$, $d_{CP}=1696$, $k=3384$;
或者, $r=305$, $d_{CP}=2168$, $k=2668$;
10 或者, $r=305$, $d_{CP}=656$, $k=3394$;
或者, $r=305$, $d_{CP}=2504$, $k=3396$;
或者, $r=306$, $d_{CP}=1296$, $k=2676$;
或者, $r=313$, $d_{CP}=1912$, $k=3194$;
或者, $r=314$, $d_{CP}=1712$, $k=3204$;
15 或者, $r=315$, $d_{CP}=1512$, $k=3214$;
或者, $r=316$, $d_{CP}=1312$, $k=3224$;
或者, $r=317$, $d_{CP}=1112$, $k=3234$;
或者, $r=317$, $d_{CP}=3128$, $k=3236$;
或者, $r=326$, $d_{CP}=320$, $k=2660$;
20 或者, $r=326$, $d_{CP}=2840$, $k=2662$;
或者, $r=327$, $d_{CP}=120$, $k=2668$;
或者, $r=327$, $d_{CP}=2640$, $k=2670$;
或者, $r=328$, $d_{CP}=2440$, $k=2678$;
或者, $r=329$, $d_{CP}=728$, $k=3356$;
25 或者, $r=329$, $d_{CP}=2744$, $k=3358$;
或者, $r=330$, $d_{CP}=528$, $k=3366$;
或者, $r=330$, $d_{CP}=2544$, $k=3368$;
或者, $r=331$, $d_{CP}=328$, $k=3376$;
或者, $r=331$, $d_{CP}=2344$, $k=3378$;
30 或者, $r=332$, $d_{CP}=128$, $k=3386$;
或者, $r=332$, $d_{CP}=2144$, $k=3388$;
或者, $r=333$, $d_{CP}=1944$, $k=3398$;
或者, $r=339$, $d_{CP}=744$, $k=3192$;
或者, $r=339$, $d_{CP}=2928$, $k=3194$;
35 或者, $r=340$, $d_{CP}=1384$, $k=3202$;
或者, $r=341$, $d_{CP}=2024$, $k=3212$;
或者, $r=342$, $d_{CP}=480$, $k=3220$;
或者, $r=342$, $d_{CP}=2664$, $k=3222$;
或者, $r=343$, $d_{CP}=1120$, $k=3230$;
40 或者, $r=343$, $d_{CP}=3304$, $k=3232$;
或者, $r=344$, $d_{CP}=1760$, $k=3240$;
或者, $r=348$, $d_{CP}=288$, $k=2662$;
或者, $r=348$, $d_{CP}=2976$, $k=2664$;
或者, $r=349$, $d_{CP}=760$, $k=2670$;
45 或者, $r=349$, $d_{CP}=3448$, $k=2672$;
或者, $r=350$, $d_{CP}=1232$, $k=2678$;

- 或者, $r=356$, $d_{CP}=704$, $k=3352$;
或者, $r=356$, $d_{CP}=2888$, $k=3354$;
或者, $r=357$, $d_{CP}=1344$, $k=3362$;
或者, $r=357$, $d_{CP}=3528$, $k=3364$;
5 或者, $r=358$, $d_{CP}=1984$, $k=3372$;
或者, $r=359$, $d_{CP}=440$, $k=3380$;
或者, $r=359$, $d_{CP}=2624$, $k=3382$;
或者, $r=360$, $d_{CP}=1080$, $k=3390$;
或者, $r=360$, $d_{CP}=3264$, $k=3392$;
10 或者, $r=361$, $d_{CP}=1720$, $k=3400$;
或者, $r=365$, $d_{CP}=1592$, $k=3192$;
或者, $r=366$, $d_{CP}=720$, $k=3200$;
或者, $r=366$, $d_{CP}=3072$, $k=3202$;
或者, $r=367$, $d_{CP}=2200$, $k=3210$;
15 或者, $r=368$, $d_{CP}=1328$, $k=3218$;
或者, $r=368$, $d_{CP}=3680$, $k=3220$;
或者, $r=369$, $d_{CP}=2304$, $k=2658$;
或者, $r=369$, $d_{CP}=456$, $k=3226$;
或者, $r=369$, $d_{CP}=2808$, $k=3228$;
20 或者, $r=370$, $d_{CP}=592$, $k=2664$;
或者, $r=370$, $d_{CP}=3448$, $k=2666$;
或者, $r=370$, $d_{CP}=1936$, $k=3236$;
或者, $r=371$, $d_{CP}=1736$, $k=2672$;
或者, $r=384$, $d_{CP}=1488$, $k=3358$;
25 或者, $r=384$, $d_{CP}=3840$, $k=3360$;
或者, $r=385$, $d_{CP}=616$, $k=3366$;
或者, $r=385$, $d_{CP}=2968$, $k=3368$;
或者, $r=386$, $d_{CP}=2096$, $k=3376$;
或者, $r=387$, $d_{CP}=1224$, $k=3384$;
30 或者, $r=387$, $d_{CP}=3576$, $k=3386$;
或者, $r=388$, $d_{CP}=352$, $k=3392$;
或者, $r=388$, $d_{CP}=2704$, $k=3394$;
或者, $r=389$, $d_{CP}=1832$, $k=3402$;
或者, $r=391$, $d_{CP}=2440$, $k=2660$;
35 或者, $r=392$, $d_{CP}=1232$, $k=2666$;
或者, $r=392$, $d_{CP}=2240$, $k=3200$;
或者, $r=393$, $d_{CP}=24$, $k=2672$;
或者, $r=393$, $d_{CP}=3048$, $k=2674$;
或者, $r=393$, $d_{CP}=2040$, $k=3208$;
40 或者, $r=394$, $d_{CP}=1840$, $k=3216$;
或者, $r=395$, $d_{CP}=1640$, $k=3224$;
或者, $r=396$, $d_{CP}=1440$, $k=3232$;
或者, $r=396$, $d_{CP}=3960$, $k=3234$;
或者, $r=397$, $d_{CP}=1240$, $k=3240$;
45 或者, $r=397$, $d_{CP}=3760$, $k=3242$;
或者, $r=411$, $d_{CP}=960$, $k=3354$;

- 或者, $r=411$, $d_{CP}=3480$, $k=3356$;
或者, $r=412$, $d_{CP}=760$, $k=3362$;
或者, $r=412$, $d_{CP}=3280$, $k=3364$;
或者, $r=413$, $d_{CP}=2912$, $k=2662$;
5 或者, $r=413$, $d_{CP}=560$, $k=3370$;
或者, $r=413$, $d_{CP}=3080$, $k=3372$;
或者, $r=414$, $d_{CP}=2208$, $k=2668$;
或者, $r=414$, $d_{CP}=360$, $k=3378$;
或者, $r=414$, $d_{CP}=2880$, $k=3380$;
10 或者, $r=415$, $d_{CP}=1504$, $k=2674$;
或者, $r=415$, $d_{CP}=160$, $k=3386$;
或者, $r=415$, $d_{CP}=2680$, $k=3388$;
或者, $r=416$, $d_{CP}=2480$, $k=3396$;
或者, $r=418$, $d_{CP}=1072$, $k=3198$;
15 或者, $r=418$, $d_{CP}=3760$, $k=3200$;
或者, $r=419$, $d_{CP}=1544$, $k=3206$;
或者, $r=419$, $d_{CP}=4232$, $k=3208$;
或者, $r=420$, $d_{CP}=2016$, $k=3214$;
或者, $r=421$, $d_{CP}=2488$, $k=3222$;
20 或者, $r=422$, $d_{CP}=272$, $k=3228$;
或者, $r=422$, $d_{CP}=2960$, $k=3230$;
或者, $r=423$, $d_{CP}=744$, $k=3236$;
或者, $r=423$, $d_{CP}=3432$, $k=3238$;
或者, $r=434$, $d_{CP}=560$, $k=2656$;
25 或者, $r=434$, $d_{CP}=3920$, $k=2658$;
或者, $r=435$, $d_{CP}=360$, $k=2662$;
或者, $r=435$, $d_{CP}=3720$, $k=2664$;
或者, $r=436$, $d_{CP}=160$, $k=2668$;
或者, $r=436$, $d_{CP}=3520$, $k=2670$;
30 或者, $r=437$, $d_{CP}=3320$, $k=2676$;
或者, $r=438$, $d_{CP}=2448$, $k=3352$;
或者, $r=439$, $d_{CP}=232$, $k=3358$;
或者, $r=439$, $d_{CP}=2920$, $k=3360$;
或者, $r=440$, $d_{CP}=704$, $k=3366$;
35 或者, $r=440$, $d_{CP}=3392$, $k=3368$;
或者, $r=441$, $d_{CP}=1176$, $k=3374$;
或者, $r=441$, $d_{CP}=3864$, $k=3376$;
或者, $r=442$, $d_{CP}=1648$, $k=3382$;
或者, $r=442$, $d_{CP}=4336$, $k=3384$;
40 或者, $r=443$, $d_{CP}=2120$, $k=3390$;
或者, $r=444$, $d_{CP}=2424$, $k=3198$;
或者, $r=444$, $d_{CP}=2592$, $k=3398$;
或者, $r=445$, $d_{CP}=712$, $k=3204$;
或者, $r=445$, $d_{CP}=3568$, $k=3206$;
45 或者, $r=446$, $d_{CP}=1856$, $k=3212$;
或者, $r=447$, $d_{CP}=144$, $k=3218$;

- 或者, $r=447$, $d_{CP}=3000$, $k=3220$;
或者, $r=448$, $d_{CP}=1288$, $k=3226$;
或者, $r=448$, $d_{CP}=4144$, $k=3228$;
或者, $r=449$, $d_{CP}=2432$, $k=3234$;
5 或者, $r=456$, $d_{CP}=1032$, $k=2658$;
或者, $r=456$, $d_{CP}=4560$, $k=2660$;
或者, $r=457$, $d_{CP}=1336$, $k=2664$;
或者, $r=458$, $d_{CP}=1640$, $k=2670$;
或者, $r=459$, $d_{CP}=1944$, $k=2676$;
10 或者, $r=466$, $d_{CP}=1888$, $k=3356$;
或者, $r=466$, $d_{CP}=4744$, $k=3358$;
或者, $r=467$, $d_{CP}=176$, $k=3362$;
或者, $r=467$, $d_{CP}=3032$, $k=3364$;
或者, $r=468$, $d_{CP}=1320$, $k=3370$;
15 或者, $r=468$, $d_{CP}=4176$, $k=3372$;
或者, $r=469$, $d_{CP}=2464$, $k=3378$;
或者, $r=470$, $d_{CP}=752$, $k=3196$;
或者, $r=470$, $d_{CP}=3776$, $k=3198$;
或者, $r=470$, $d_{CP}=752$, $k=3384$;
20 或者, $r=470$, $d_{CP}=3608$, $k=3386$;
或者, $r=471$, $d_{CP}=2568$, $k=3204$;
或者, $r=471$, $d_{CP}=1896$, $k=3392$;
或者, $r=471$, $d_{CP}=4752$, $k=3394$;
或者, $r=472$, $d_{CP}=1360$, $k=3210$;
25 或者, $r=472$, $d_{CP}=4384$, $k=3212$;
或者, $r=472$, $d_{CP}=184$, $k=3398$;
或者, $r=472$, $d_{CP}=3040$, $k=3400$;
或者, $r=473$, $d_{CP}=152$, $k=3216$;
或者, $r=473$, $d_{CP}=3176$, $k=3218$;
30 或者, $r=474$, $d_{CP}=1968$, $k=3224$;
或者, $r=475$, $d_{CP}=760$, $k=3230$;
或者, $r=475$, $d_{CP}=3784$, $k=3232$;
或者, $r=476$, $d_{CP}=2576$, $k=3238$;
或者, $r=478$, $d_{CP}=1840$, $k=2660$;
35 或者, $r=479$, $d_{CP}=2648$, $k=2666$;
或者, $r=480$, $d_{CP}=3456$, $k=2672$;
或者, $r=481$, $d_{CP}=568$, $k=2676$;
或者, $r=481$, $d_{CP}=4264$, $k=2678$;
或者, $r=493$, $d_{CP}=184$, $k=3352$;
40 或者, $r=493$, $d_{CP}=3208$, $k=3354$;
或者, $r=494$, $d_{CP}=2000$, $k=3360$;
或者, $r=494$, $d_{CP}=5024$, $k=3362$;
或者, $r=495$, $d_{CP}=792$, $k=3366$;
或者, $r=495$, $d_{CP}=3816$, $k=3368$;
45 或者, $r=496$, $d_{CP}=1936$, $k=3196$;
或者, $r=496$, $d_{CP}=2608$, $k=3374$;

- 或者, $r=497$, $d_{CP}=1232$, $k=3202$;
 或者, $r=497$, $d_{CP}=4424$, $k=3204$;
 或者, $r=497$, $d_{CP}=1400$, $k=3380$;
 或者, $r=497$, $d_{CP}=4424$, $k=3382$;
 5 或者, $r=498$, $d_{CP}=528$, $k=3208$;
 或者, $r=498$, $d_{CP}=3720$, $k=3210$;
 或者, $r=498$, $d_{CP}=192$, $k=3386$;
 或者, $r=498$, $d_{CP}=3216$, $k=3388$;
 或者, $r=499$, $d_{CP}=1672$, $k=2656$;
 10 或者, $r=499$, $d_{CP}=3016$, $k=3216$;
 或者, $r=499$, $d_{CP}=2008$, $k=3394$;
 或者, $r=499$, $d_{CP}=5032$, $k=3396$;
 或者, $r=500$, $d_{CP}=2984$, $k=2662$;
 或者, $r=500$, $d_{CP}=2312$, $k=3222$;
 15 或者, $r=500$, $d_{CP}=800$, $k=3400$;
 或者, $r=500$, $d_{CP}=3824$, $k=3402$;
 或者, $r=501$, $d_{CP}=432$, $k=2666$;
 或者, $r=501$, $d_{CP}=4296$, $k=2668$;
 或者, $r=501$, $d_{CP}=1608$, $k=3228$;
 20 或者, $r=501$, $d_{CP}=4800$, $k=3230$;
 或者, $r=502$, $d_{CP}=1744$, $k=2672$;
 或者, $r=502$, $d_{CP}=904$, $k=3234$;
 或者, $r=502$, $d_{CP}=4096$, $k=3236$;
 或者, $r=503$, $d_{CP}=3056$, $k=2678$ 。
 25 75. 根据权利要求 70 至 74 中任一项所述的数据处理装置, 其特征在于, $d_{scr} = 336 \times k$ 。
 76. 根据权利要求 70 至 75 中任一项所述的数据处理装置, 其特征在于, 所述第二数据处理单元具体用于:
 对所述第一数据中每 r/p 行总共 $r \times q/p$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特,
 $d_{CRC} = 32 \times p$, r 能被 p 整除, p 为大于 1 的整数;
 30 或者,
 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数 r_{F0} 大于整数 r_{F1} 。
 35 77. 根据权利要求 70 至 76 中任一项所述的数据处理装置, 其特征在于, $q=2056$ 。
 78. 根据权利要求 77 所述的数据处理装置, 其特征在于, 所述第二数据处理单元具体用于:
 对所述第一数据中前 $r_{F0} \times (p-1)$ 行中的每 r_{F0} 行共 $q \times r_{F0}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 并对所述第一数据中最后 r_{F1} 行共 $q \times r_{F1}$ 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特, 其中, $d_{CRC} = 32 \times p$, r 不能被 p 整除, p 为大于 1 的整数, $r_{F0} \times (p-1) + r_{F1} = r$, 整数
 40 r_{F0} 大于整数 r_{F1} 。
 79. 根据权利要求 77 或 78 所述的数据处理装置, 其特征在于, $r=435$ 。
 80. 根据权利要求 79 所述的数据处理装置, 其特征在于, $p=22$, $r_{F0}=20$, $r_{F1}=15$, 所述第二数据处理单元具体用于:
 对所述第一数据中前 420 行中的每 20 行共 41120 个比特进行 CRC-32 校验以添加长度为 32 比特的
 45 校验比特, 并对所述第一数据中最后 15 行共 30840 个比特进行 CRC-32 校验以添加长度为 32 比特的校验比特。

81. 根据权利要求 79 或 80 所述的数据处理装置, 其特征在于, $d_{CP} = 3432$, $k = 2672$, 或者, $d_{CP} = 744$, $k = 2664$ 。

82. 根据权利要求 77 或 78 所述的数据处理装置, 其特征在于, $r=546$, $d_{CP}=1008$, $k=3344$;
或者, $r=546$, $d_{CP}=3696$, $k=3352$;
5 或者, $r=547$, $d_{CP}=1640$, $k=3352$;
或者, $r=547$, $d_{CP}=4328$, $k=3360$;
或者, $r=548$, $d_{CP}=2272$, $k=3360$;
或者, $r=548$, $d_{CP}=4960$, $k=3368$;
或者, $r=549$, $d_{CP}=216$, $k=3360$;
10 或者, $r=549$, $d_{CP}=2904$, $k=3368$;
或者, $r=549$, $d_{CP}=5592$, $k=3376$;
或者, $r=550$, $d_{CP}=848$, $k=3368$;
或者, $r=550$, $d_{CP}=3536$, $k=3376$;
或者, $r=551$, $d_{CP}=1480$, $k=3376$;
15 或者, $r=551$, $d_{CP}=4168$, $k=3384$;
或者, $r=552$, $d_{CP}=2112$, $k=3384$;
或者, $r=552$, $d_{CP}=4800$, $k=3392$;
或者, $r=553$, $d_{CP}=56$, $k=3384$;
或者, $r=553$, $d_{CP}=2744$, $k=3392$;
20 或者, $r=553$, $d_{CP}=5432$, $k=3400$;
或者, $r=554$, $d_{CP}=688$, $k=3392$;
或者, $r=554$, $d_{CP}=3376$, $k=3400$;
或者, $r=555$, $d_{CP}=1320$, $k=3400$;
或者, $r=555$, $d_{CP}=4008$, $k=3408$;
25 或者, $r=556$, $d_{CP}=1952$, $k=3408$;
或者, $r=556$, $d_{CP}=4640$, $k=3416$;
或者, $r=557$, $d_{CP}=2584$, $k=3416$;
或者, $r=557$, $d_{CP}=5272$, $k=3424$;
或者, $r=558$, $d_{CP}=528$, $k=3416$;
30 或者, $r=558$, $d_{CP}=3216$, $k=3424$;
或者, $r=559$, $d_{CP}=1160$, $k=3424$;
或者, $r=559$, $d_{CP}=3848$, $k=3432$;
或者, $r=560$, $d_{CP}=1792$, $k=3432$;
或者, $r=560$, $d_{CP}=4480$, $k=3440$;
35 或者, $r=561$, $d_{CP}=2424$, $k=3440$;
或者, $r=561$, $d_{CP}=5112$, $k=3448$;
或者, $r=562$, $d_{CP}=368$, $k=3440$;
或者, $r=562$, $d_{CP}=3056$, $k=3448$;
或者, $r=562$, $d_{CP}=5744$, $k=3456$ 。
40 83. 根据权利要求 77 或 78 所述的数据处理装置, 其特征在于, $r=520$, $d_{CP}=704$, $k=3184$;
或者, $r=520$, $d_{CP}=3392$, $k=3192$;
或者, $r=520$, $d_{CP}=6080$, $k=3200$;
或者, $r=521$, $d_{CP}=1336$, $k=3192$;
或者, $r=521$, $d_{CP}=4024$, $k=3200$;
45 或者, $r=522$, $d_{CP}=1968$, $k=3200$;
或者, $r=522$, $d_{CP}=4656$, $k=3208$;

- 或者, $r=523$, $d_{CP}=2600$, $k=3208$;
或者, $r=523$, $d_{CP}=5288$, $k=3216$;
或者, $r=524$, $d_{CP}=544$, $k=3208$;
或者, $r=524$, $d_{CP}=3232$, $k=3216$;
5 或者, $r=525$, $d_{CP}=1176$, $k=3216$;
或者, $r=525$, $d_{CP}=3864$, $k=3224$;
或者, $r=526$, $d_{CP}=1808$, $k=3224$;
或者, $r=526$, $d_{CP}=4496$, $k=3232$;
或者, $r=527$, $d_{CP}=2440$, $k=3232$;
10 或者, $r=527$, $d_{CP}=5128$, $k=3240$;
或者, $r=528$, $d_{CP}=384$, $k=3232$;
或者, $r=528$, $d_{CP}=3072$, $k=3240$;
或者, $r=528$, $d_{CP}=5760$, $k=3248$;
或者, $r=529$, $d_{CP}=1016$, $k=3240$;
15 或者, $r=529$, $d_{CP}=3704$, $k=3248$;
或者, $r=530$, $d_{CP}=1648$, $k=3248$;
或者, $r=530$, $d_{CP}=4336$, $k=3256$;
或者, $r=531$, $d_{CP}=2280$, $k=3256$;
或者, $r=531$, $d_{CP}=4968$, $k=3264$;
20 或者, $r=532$, $d_{CP}=2912$, $k=3264$;
或者, $r=532$, $d_{CP}=224$, $k=3256$;
或者, $r=533$, $d_{CP}=856$, $k=3264$;
或者, $r=533$, $d_{CP}=3544$, $k=3272$;
或者, $r=534$, $d_{CP}=1488$, $k=3272$;
25 或者, $r=534$, $d_{CP}=4176$, $k=3280$;
或者, $r=535$, $d_{CP}=2120$, $k=3280$;
或者, $r=535$, $d_{CP}=4808$, $k=3288$ 。

84. 根据权利要求 70 至 83 中任一项所述的数据处理装置, 其特征在于, 所述第四比特集合包括 m_0 个第一比特子集, 所述 m_0 为大于 1 的整数, 所述数据处理装置还包括第一交织单元;

- 30 所述第一交织单元用于: 对所述第四比特集合进行第一交织处理得到第五比特集合, 所述第五比特集合包括 m_0 个第二比特子集, 每个所述第二比特子集包括 256 个比特, 每个所述第二比特子集中的其中 128 个比特来自所述第二比特集合, 每个所述第二比特子集中的另外 128 个比特来自所述第三比特集合和/或所述 FEC 编码的校验比特。

- 35 85. 根据权利要求 84 所述的数据处理装置, 其特征在于, 每个所述第二比特子集包括连续的 256 个比特。

86. 根据权利要求 84 所述的数据处理装置, 其特征在于, 每个所述第二比特子集呈现为方阵。

87. 根据权利要求 84 至 86 中任一项所述的数据处理装置, 其特征在于, 所述第二比特子集的 256 个比特分布为 16 行 16 列, 所述第二比特子集中的比特满足以下的一项或多项:

- 40 所述第二比特子集中第 0 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
所述第二比特子集中第 1 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
所述第二比特子集中第 2 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
所述第二比特子集中第 3 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;
所述第二比特子集中第 4 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;
所述第二比特子集中第 5 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;
45 所述第二比特子集中第 6 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;
所述第二比特子集中第 7 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;

所述第二比特子集中第 8 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;

所述第二比特子集中第 9 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;

所述第二比特子集中第 10 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;

所述第二比特子集中第 11 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合;

5 所述第二比特子集中第 12 行第 2, 3, 6, 7, 10, 11, 14, 15 列的比特来自所述第二比特集合;

所述第二比特子集中第 13 行第 0, 3, 4, 7, 8, 11, 12, 15 列的比特来自所述第二比特集合;

所述第二比特子集中第 14 行第 0, 1, 4, 5, 8, 9, 12, 13 列的比特来自所述第二比特集合;

所述第二比特子集中第 15 行第 1, 2, 5, 6, 9, 10, 13, 14 列的比特来自所述第二比特集合。

88. 根据权利要求 84 至 87 中任一项所述的数据处理装置, 其特征在于, 所述第四比特集合中
10 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中 $m_0 \times 256$ 个比特分布为 32 行 $m_0 \times 8$ 列, 所述第五比特集合中第 r_1 行的 $m_0 \times 8$ 个比特来自所述第四比特集合中第 r_0 行的 $m_0 \times 8$ 个比特, 其中,
 $0 \leq r_0 < 32, 0 \leq r_1 < 32$ 。

89. 根据权利要求 88 所述的数据处理装置, 其特征在于, $r_0 = r_1$ 。

90. 根据权利要求 70 至 89 任一项所述的数据处理装置, 其特征在于, 所述第四比特集合包括 m_0 个
15 第一比特子集, $m_0=12$ 或 16。

91. 根据权利要求 84 至 90 中任一项所述的数据处理装置, 其特征在于, 所述数据处理装置还包括第二交织单元、符号映射单元、偏振划分单元和数字信号处理 DSP 成帧单元;

所述第二交织单元用于: 对 L 条第一比特流中每两条第一比特流进行第二交织处理得到共 L/2 条第二比特流, 每条所述第一比特流包括多个所述第五比特集合, L 为大于 0 的偶数;

20 所述符号映射单元和所述偏振划分单元用于: 对 L/2 条所述第二比特流进行符号映射和偏振划分得到一条双偏振符号流, 其中, 所述符号映射和所述偏振划分将 t 个比特映射为 1 个双偏振符号, 所述 t 为大于 0 的整数;

所述 DSP 成帧单元用于: 对所述双偏振符号流进行 DSP 成帧。

92. 根据权利要求 91 所述的数据处理装置, 其特征在于, 所述双偏振符号中的幅度位比特来自所述
25 第二比特集合。

93. 根据权利要求 92 所述的数据处理装置, 其特征在于, $t=8$, 所述双偏振符号为双偏振 DP-16QAM 符号。

94. 根据权利要求 93 所述的数据处理装置, 其特征在于, 所述 DSP 成帧单元具体用于: 将每 172032 个双偏振 DP-16QAM 符号进行成帧处理得到一个 DSP 超帧, 所述超帧包含 175104 个双偏振符号。

30 95. 根据权利要求 94 所述的数据处理装置, 其特征在于, 所述第一处理单元具体用于: 从所述数据帧中获取 F_{Frame} 个第一数据, 以得到 $F_{\text{Frame}} \times r \times q$ 个比特, F_{Frame} 为大于 0 的整数;

$$d_{\text{scr}} \times F_{\text{Frame}} = 336 \times k \times F_{\text{DSP}};$$

其中, F_{DSP} 表示所述 DSP 超帧的数量, F_{DSP} 个 DSP 超帧由 F_{Frame} 个第一数据经过处理得到, F_{DSP} 为大于 0 的整数。

35 96. 根据权利要求 95 所述的数据处理装置, 其特征在于, $F_{\text{DSP}} > 1$, 所述 F_{DSP} 个 DSP 超帧中第一 DSP 超帧中的连续多个符号为第一标识, 所述 F_{DSP} 个 DSP 超帧中除所述第一 DSP 超帧外的每个 DSP 超帧中的连续多个符号为第二标识。

97. 一种数据处理装置, 其特征在于, 包括: 第一概率星座整形 PCS 单元、第一前向纠错 FEC 编码单元、第二 PCS 单元、第二 FEC 单元、交织处理单元、符号映射单元和偏振划分单元;

40 所述第一 PCS 单元用于: 对第一组 k 个比特中的第一比特集合进行第一 PCS 处理得到第二比特集合, 所述第二比特集合包括 $m_0 \times 128$ 个比特, k 为大于 1 的整数, 所述 m_0 为大于 1 的整数;

所述第一 FEC 编码单元用于: 对第二比特集合和所述第一组 k 个比特中除所述第一比特集合外的第三比特集合进行第一前向纠错 FEC 编码得到第四比特集合, 所述第四比特集合包括 m_0 个第一比特方阵, 每个所述第一比特方阵包括分布为 16 行 16 列的共 256 个比特;

45 所述第二 PCS 单元用于: 对第二组 k 个比特中的第五比特集合进行第二 PCS 处理得到第六比特集合;

所述第二 FEC 编码单元用于: 对第六比特集合和所述第二组 k 个比特中除所述第五比特集合外的第

七比特集合进行第二 FEC 编码得到第八比特集合, 所述第八比特集合包括 m_0 个第二比特方阵, 每个所述第二比特方阵包括分布为 16 行 16 列的共 256 个比特;

所述交织处理单元用于: 将 21 个所述第四比特集合和 21 个所述第八比特集合输入交织缓存, 其中, 所述交织缓存包括分布为 84 行 m_1 列的共 $84 \times m_1$ 个缓存单元, $m_0 = m_1 \times 2$, 每个所述缓存单元用于缓存 16 行 16 列共 256 个比特, $42 \times m_1$ 个所述第一比特方阵输入所述交织缓存的偶数行, $42 \times m_1$ 个所述第二比特方阵输入所述交织缓存的奇数行, 所述交织缓存包括第一缓存子集和第二缓存子集, 所述第一缓存子集包括 $42 \times m_1$ 个所述缓存单元, 所述第二缓存子集包括 $42 \times m_1$ 个所述缓存单元, 输入所述第一缓存子集的比特来自所述第二比特集合和所述第六比特集合, 输入所述第二缓存子集的比特来自所述第三比特集合、所述第七比特集合、第一 FEC 编码的校验比特和第二 FEC 编码的校验比特;

所述符号映射单元和所述偏振划分单元用于: 对所述交织缓存中每 8 个比特进行符号映射和偏振划分得到 1 个双偏振 DP-16QAM 符号, 所述 DP-16QAM 符号在目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个幅度位比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 DP-16QAM 符号在所述目标偏振方向上的 16QAM 符号所对应的 4 个比特中的 2 个符号位比特来自所述第二缓存子集的第二缓存单元中的一列。

98. 根据权利要求 97 所述的数据处理装置, 其特征在于, 所述 8 个比特中的其中 4 个比特来自所述第一缓存子集的第一缓存单元中的一列, 所述 8 个比特中的另外 4 个比特来自所述第二缓存子集的第二缓存单元中的一列。

99. 根据权利要求 97 或 98 所述的数据处理装置, 其特征在于, 所述 m_0 个第一比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第一比特方阵和第 1 行第 m_2 列的第一比特方阵来自所述第二比特集合, 所述 m_0 个第二比特方阵分布为 2 行 m_1 列, 其中, 第 0 行第 m_2 列的第二比特方阵和第 1 行第 m_2 列的第二比特方阵来自所述第六比特集合, $0 \leq m_2 < m_1/2$ 。

100. 根据权利要求 99 所述的数据处理装置, 其特征在于, $m_0 = 16$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 2 列和第 3 列中的共 336 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 4 列、第 5 列、第 6 列和第 7 列中的共 336 个缓存单元;

或者, $m_0 = 12$, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列和第 2 列中的共 252 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 3 列、第 4 列和第 5 列中的共 252 个缓存单元。

101. 根据权利要求 97 或 98 所述的数据处理装置, 其特征在于, $m_0 = 16$, $m_1 = 8$;

所述 m_0 个第一比特方阵分布为 2 行 m_1 列, 第 0 行第 0 列的第一比特方阵、第 0 行第 1 列的第一比特方阵、第 0 行第 4 列的第一比特方阵、第 0 行第 5 列的第一比特方阵、第 1 行第 0 列的第一比特方阵、第 1 行第 1 列的第一比特方阵、第 1 行第 4 列的第一比特方阵和第 1 行第 5 列的第一比特方阵来自所述第二比特集合;

所述 m_0 个第二比特方阵分布为 2 行 m_1 列, 第 0 行第 0 列的第二比特方阵、第 0 行第 1 列的第二比特方阵、第 0 行第 4 列的第二比特方阵、第 0 行第 5 列的第二比特方阵、第 1 行第 0 列的第二比特方阵、第 1 行第 1 列的第二比特方阵、第 1 行第 4 列的第二比特方阵和第 1 行第 5 列的第二比特方阵来自所述第六比特集合。

102. 根据权利要求 101 所述的数据处理装置, 其特征在于, 所述第一缓存子集包括所述交织缓存中第 0 列、第 1 列、第 4 列和第 5 列中的共 336 个缓存单元, 所述第二缓存子集包括所述交织缓存中第 2 列、第 3 列、第 6 列和第 7 列中的共 336 个缓存单元。

103. 一种数据处理装置, 其特征在于, 包括: 处理器和收发器, 处理器用于执行如权利要求 1 至 51 中任一项所述的方法, 所述收发器用于执行数据收发的操作。

104. 一种通信系统, 其特征在于, 包括: 发端数据处理装置和收端数据处理装置, 所述发端数据处理装置用于执行如权利要求 1-51 中任一项所述的方法, 所述发端数据处理装置用于向所述收端数据处理装置发送数据流。