



(12)发明专利

(10)授权公告号 CN 103795496 B

(45)授权公告日 2017. 10. 20

(21)申请号 201410019336.2

(22)申请日 2008.07.11

(65)同一申请的已公布的文献号

申请公布号 CN 103795496 A

(43)申请公布日 2014.05.14

(30)优先权数据

60/929,790 2007.07.12 US

12/076,777 2008.03.21 US

(62)分案原申请数据

200880022897.0 2008.07.11

(73)专利权人 三星电子株式会社

地址 韩国京畿道水原市

(72)发明人 皮周月 法罗克·坎

(74)专利代理机构 北京铭硕知识产权代理有限公司 11286

代理人 张云珠 张军

(51)Int.Cl.

H04L 1/00(2006.01)

H04L 1/18(2006.01)

H03M 13/09(2006.01)

(56)对比文件

CN 1661951 A,2005.08.31,

CN 1292188 A,2001.04.18,

US 2004193993 A1,2004.09.03,

审查员 徐佳

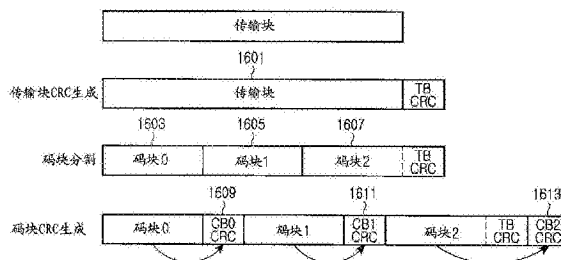
权利要求书2页 说明书16页 附图23页

(54)发明名称

在通信系统中为多个码块计算CRC的方法和
设备

(57)摘要

一种生成循环冗余校验的方法和电路。所述方法为具有多个信息位的传输块计算多个循环冗余校验。为包括多个信息位的传输块计算传输块CRC。将包括传输块CRC的传输块分割为多个子集,为所述多个子集计算多个循环冗余校验。基于信息位的子集计算所述多个循环冗余校验中的至少一个循环冗余校验。另外,可基于所有信息位计算传输块循环冗余校验。



1. 一种用于通信的方法,所述方法包括:
为传输块的所有信息位产生传输块循环冗余校验CRC;
当包括传输块CRC的传输块的位的数量大于预定值时,将传输块分割为多个码块,其中,所述多个码块中的一个码块包括传输块CRC;
为通过分割传输块产生的所述多个码块,产生多个码块CRC;
对所述多个码块中的每个码块和分别与所述多个码块中的每个码块相应地产生的所述多个码块CRC中的每个码块CRC进行联合编码;
发送所述多个码块和所述多个码块CRC。
2. 如权利要求1所述的方法,其中,联合编码的步骤包括:
通过特定类型的前向纠错码来对所述多个码块中的每个码块和所述多个码块CRC中的每个码块CRC进行联合编码,
其中,所述特定类型的前向纠错码是turbo码。
3. 如权利要求1所述的方法,其中,产生所述多个码块CRC的步骤还包括:
分别将所述多个码块CRC附着到所述多个码块。
4. 如权利要求1所述的方法,还包括:
将传输块CRC附着到信息位的末端,其中,通过分割传输块产生的所述多个码块中的至少一个码块包括传输块CRC。
5. 如权利要求1所述的方法,其中,传输块CRC包括L个校验位。
6. 如权利要求1所述的方法,其中,所述多个码块CRC中的每个码块CRC包括L个校验位。
7. 如权利要求1所述的方法,还包括:通过用户设备在上行链路通信传输中发送所述多个码块和所述多个码块CRC。
8. 如权利要求1所述的方法,还包括:通过基站在下行链路通信传输中发送所述多个码块和所述多个码块CRC。
9. 如权利要求1所述的方法,其中,传输块CRC在所述多个码块中的一个码块中被发送。
10. 一种通信设备,包括:
第一循环冗余校验CRC产生器,被配置为为传输块的所有信息位产生传输块循环冗余校验CRC;
分割单元,被配置为当包括传输块CRC的传输块的位的数量大于预定值时,将传输块分割为多个码块,其中,所述多个码块中的一个码块包括传输块CRC;
第二CRC产生器,被配置为为通过分割传输块产生的所述多个码块,产生多个码块CRC;
编码器,被配置为对所述多个码块中的每个码块和分别与所述多个码块中的每个码块相应地产生的所述多个码块CRC中的每个码块CRC进行联合编码;
发送机,被配置为发送所述多个码块和所述多个码块CRC。
11. 如权利要求10所述的设备,其中,编码器还被配置为通过特定类型的前向纠错码来对所述多个码块中的每个码块和所述多个码块CRC中的每个码块CRC进行联合编码,
其中,所述特定类型的前向纠错码是turbo码。
12. 如权利要求10所述的设备,其中,第二CRC产生器还被配置为分别将所述多个码块CRC附着到所述多个码块。
13. 如权利要求10所述的设备,其中,传输块CRC被附着到信息位的末端,其中,通过分

割传输块产生的所述多个码块中的至少一个码块包含传输块CRC。

14. 如权利要求10所述的设备,其中,传输块CRC包括L个校验位。

15. 如权利要求10所述的设备,其中,所述多个码块CRC中的每个码块CRC包括L个校验位。

16. 如权利要求10所述的设备,其中,通过用户设备在上行链路通信传输中发送所述多个码块和所述多个码块CRC。

17. 如权利要求10所述的设备,还包括:通过基站在下行链路通信传输中发送所述多个码块和所述多个码块CRC。

18. 如权利要求10所述的设备,其中,传输块CRC在所述多个码块中的一个码块中被发送。

19. 一种用于通信的方法,所述方法包括:

为传输块的所有信息位产生传输块循环冗余校验CRC;

如果包括产生的传输块CRC的传输块的位的数量大于预定的位的数量时,将传输块分割为多个码块,其中,所述多个码块中的一个码块包括传输块CRC;

为通过分割传输块产生的所述多个码块,产生多个码块CRC;

对所述多个码块中的每个码块和分别与所述多个码块中的每个码块相应地产生的所述多个码块CRC中的每个码块CRC进行联合编码;

发送所述多个码块和所述多个码块CRC,

其中,所述多个码块CRC中的至少一个码块CRC是基于所述多个码块中的一个码块产生的。

20. 如权利要求19所述的方法,其中,联合编码的步骤包括:

通过特定类型的前向纠错码来对所述多个码块中的每个码块和所述多个码块CRC中的每个码块CRC进行联合编码,

其中,所述特定类型的前向纠错码是turbo码。

21. 如权利要求19所述的方法,其中,产生所述多个码块CRC的步骤还包括:

分别将所述多个码块CRC附着到所述多个码块。

22. 如权利要求19所述的方法,还包括:

将传输块CRC附着到信息位的末端,其中,通过分割传输块产生的所述多个码块中的至少一个码块包含传输块CRC。

23. 如权利要求19所述的方法,其中,传输块CRC包括L个校验位。

24. 如权利要求19所述的方法,其中,所述多个码块CRC中的每个码块CRC包括L个校验位。

25. 如权利要求19所述的方法,还包括:通过用户设备在上行链路通信传输中发送所述多个码块和所述多个码块CRC。

26. 如权利要求19所述的方法,还包括:通过基站在下行链路通信传输中发送所述多个码块和所述多个码块CRC。

27. 如权利要求19所述的方法,其中,传输块CRC在所述多个码块中的一个码块中被发送。

在通信系统中为多个码块计算CRC的方法和设备

[0001] 本申请是向中国知识产权局提交的申请日为2008年7月11日、申请号为200880022897.0、发明名称为“在通信系统中为多个码块计算CRC的方法和设备”的申请的分案申请。

技术领域

[0002] 本发明涉及一种为多个码块生成循环冗余校验的方法和设备。

背景技术

[0003] 无线通信系统通常包括多个基站和多个移动站,而单个基站经常与一组移动站通信。从基站到移动站的传输称为下行链路通信。类似地,从移动站到基站的传输称为上行链路通信。基站和移动站均可以采用发送和接收无线电波信号的多个天线。无线电波信号可以是正交频分复用 (OFDM) 信号或码分多址 (CDMA) 信号。移动站可以是PDA、膝上型计算机或手持装置。

[0004] 在第三代合作伙伴计划长期演进 (3GPP LTE) 系统中,当传输块 (transport block) 较大时,该传输块被分割为多个码块 (code block) 从而可生成多个编码的包,由于诸如能够并行处理或者流水线执行以及功耗和硬件复杂度之间的柔性权衡的好处,这样做有利。

[0005] 在当前的高速数据共享信道 (HS-DSCH) 设计中,为整个传输块仅生成一个24位 (bit) 的循环冗余校验 (CRC), 以便对该块进行错误检测。如果生成多个码块并且在一个传输时间间隔 (TTI) 中发送所述多个码块,则接收机可能将一些码块正确解码,而不能将其它码块正确解码。在这种情况下,由于用于传输块的CRC将证明有误,所以接收机将向发送机反馈非应答 (NAK)。

发明内容

[0006] 因此,本发明的一方面在于提供在通信系统中为多个码块生成循环冗余校验的改进的方法和设备。

[0007] 本发明的另一方面在于提供一种在通信系统中用于错误检测的改进的方法和设备。

[0008] 根据本发明的一方面,提供一种用于通信的方法。为包括多个信息位的传输块计算传输块CRC。将包括传输块CRC的传输块分割为多个子集,为所述多个子集计算多个循环冗余校验。

[0009] 将所述多个循环冗余校验和所述多个信息位从第一节点发送到第二节点。

[0010] 响应于接收的所述多个循环冗余校验和所述多个信息位,在第二节点处理所述多个循环冗余校验和所述多个信息位。

[0011] 可通过特定类型的前向纠错码,如turbo码来对信息位的所述子集进行联合编码。

[0012] 可对信息位的所述子集以及基于信息位的所述子集计算的至少一个循环冗余校

验进行联合编码。

[0013] 可为信息位的第一子集计算第一循环冗余校验,可为信息位的第二子集计算第二循环冗余校验。

[0014] 信息位的第一子集和信息位的第二子集可彼此重叠。

[0015] 可选地,信息位的第一子集和信息位的第二子集可彼此分开。

[0016] 可选地,信息位的第二子集可包括信息位的第一子集。

[0017] 可基于所有信息位计算所述多个循环冗余校验中的至少一个循环冗余校验。

[0018] 根据本发明的另一方面,提供一种用于通信的方法。信息位的至少一个传输块被分割位多个码块。为所述多个码块计算多个码块循环冗余校验,其中,基于相应码块计算至少一个码块循环冗余校验。将所述多个码块和所述多个码块循环冗余校验从第一节点发送到第二节点。

[0019] 可利用特定类型的前向纠错码对从所述多个码块中选择的码块中的位进行联合编码。在这种情况下,基于联合编码的码块计算码块循环冗余校验。

[0020] 可基于所述多个码块中的相应一个码块计算所述多个码块循环冗余校验中的每一个。

[0021] 可基于所述多个码块中的至少相应一个码块计算所述多个码块循环冗余校验中的每一个。

[0022] 可在分割传输块之前,基于传输块计算传输块循环冗余校验。

[0023] 所述多个码块可包括从其没有生成码块循环冗余校验的至少一个码块。

[0024] 可基于所述多个码块中的所有码块计算至少一个码块循环冗余校验。

[0025] 根据本发明另一方面,提供一种在数据通信中生成循环冗余校验的电路,具有:输入端口,用于接收信息数据;输出端口,用于输出信息数据和循环冗余校验;线性反馈移位寄存器单元,在输入端口和输出端口之间通信连接,包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据;循环冗余校验寄存器单元,在输入端口和线性反馈移位寄存器单元之间通信连接,包括L个循环冗余校验寄存器;第一开关,在输入端口和循环冗余校验寄存器单元之间通信连接;第二开关,通信连接在线性反馈移位寄存器单元的反馈回路;第三开关,在线性反馈移位寄存器单元和循环冗余校验寄存器单元之间通信连接;第四开关,在输入端口、线性反馈移位寄存器单元和输出端口之间通信连接,具有使输入端口与输出端口连接的第一位置以及使线性反馈移位寄存器单元与输出端口连接的第二位置。

[0026] 可将线性反馈移位寄存器单元和循环冗余校验寄存器单元初始化为全零状态。可设置第一开关以将输入端口连接到线性反馈移位寄存器单元。可设置第二开关以连接线性反馈移位寄存器单元的反馈回路。可设置第三开关以使线性反馈移位寄存器单元与循环冗余校验寄存器单元之间断开连接。可将第四开关设置到第一位置,以使输入端口与输出端口连接。可经输入端口接收信息数据的码块。可设置第一开关以使输入端口与线性反馈移位寄存器单元之间断开连接。可设置第二开关以断开与线性反馈移位寄存器单元的反馈回路的连接。可设置第三开关,以使线性反馈移位寄存器单元与循环冗余校验寄存器单元连接。可将第四开关设置到第二位置,以使线性反馈移位寄存器单元与输出端口连接。可使线性反馈移位寄存器单元移位L次,以获得用于码块的循环冗余校验。

[0027] 根据本发明的另一方面,提供一种在数据通信中生成循环冗余校验的电路,所述电路包括:输入端口,用于接收信息数据;输出端口,用于输出信息数据和循环冗余校验;线性反馈移位寄存器单元,在输入端口和输出端口之间通信连接,包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据;L个状态寄存器,通信连接到L个移位寄存器中的相应的移位寄存器,以将数据值写入L个移位寄存器以及从L个移位寄存器读取数据值;第一开关,在输入端口和线性反馈移位寄存器单元之间通信连接;第二开关,通信连接在线性反馈移位寄存器单元的反馈回路;第三开关,在输入端口、线性反馈移位寄存器单元和输出端口之间通信连接,具有将输入端口与输出端口连接的第一位置以及使线性反馈移位寄存器单元与输出端口连接的第二位置。

[0028] 可将线性反馈移位寄存器单元和状态寄存器初始化为全零状态。可设置第一开关以将输入端口连接到线性反馈移位寄存器单元。可设置第二开关以连接线性反馈移位寄存器单元的反馈回路。可将第三开关设置到第一位置,以使输入端口与输出端口连接。可经输入端口接收信息数据的码块。将线性反馈移位寄存器单元中的L个移位寄存器中的数据值分别写入相应状态寄存器。可设置第一开关以使输入端口与线性反馈移位寄存器单元断开连接。可设置第二开关以断开与线性反馈移位寄存器单元的反馈回路的连接。可将第三开关设置到第二位置,以使线性反馈移位寄存器单元与输出端口连接。可使线性反馈移位寄存器单元移位L次,以获得用于码块的循环冗余校验。然后,可设置第一开关以将输入端口连接到线性反馈移位寄存器单元;可设置第二开关以连接线性反馈移位寄存器单元的反馈回路;可将第三开关设置到第一位置,以使输入端口与输出端口连接。将状态寄存器中的数据值分别写入线性反馈移位寄存器单元中的相应移位寄存器中。

[0029] 根据本发明的另一方面,提供一种在数据通信中生成循环冗余校验的电路,所述电路包括:输入端口,用于接收信息数据;输出端口,用于输出信息数据和循环冗余校验;第一线性反馈移位寄存器单元,在输入端口和输出端口之间通信连接,包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据;第二线性反馈移位寄存器单元,与第一线性反馈移位寄存器单元并联地通信连接在输出端口和输出端口之间,包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据;第一开关,在输入端口与第一线性反馈移位寄存器单元和第二线性反馈移位寄存器单元之间的公共节点之间通信连接;第二开关,通信连接在第一线性反馈移位寄存器单元的反馈回路;第三开关,在输入端口、第一线性反馈移位寄存器单元和第二线性反馈移位寄存器单元之间的所述公共节点、以及输出端口之间通信连接,具有将输入端口与输出端口连接的第一位置、使第一线性反馈移位寄存器单元与输出端口连接的第二位置以及使第二线性反馈移位寄存器单元与输出端口连接的第三位置;第四开关,通信连接在第二线性反馈移位寄存器单元的反馈回路。

[0030] 可将第一和第二线性反馈移位寄存器单元以及循环冗余校验寄存器单元初始化为全零状态。可设置第一开关,以将输入端口与第一和第二线性反馈移位寄存器单元之间的公共节点连接;可设置第二开关以连接第一线性反馈移位寄存器单元的反馈回路;可将第三开关设置到第一位置,以使输入端口与输出端口连接;可设置第四开关,以连接第二线性反馈移位寄存器单元的反馈回路。可经输入端口接收信息数据的码块。确定接收的码块是不是信息数据的最后一个码块。当接收的码块不是信息数据的最后一个码块时,可设置

第一开关以使输入端口与线性反馈移位寄存器单元断开连接;可设置第二开关以断开与线性反馈移位寄存器单元的反馈回路的连接;可将第三开关设置到第二位置,以使第一线性反馈移位寄存器单元与输出端口连接。可使第一线性反馈移位寄存器单元移位L次,以获得用于码块的循环冗余校验。

[0031] 当接收的信息数据的码块是信息数据的最后一个码块时,可将第三开关设置到第三位置,以将第二线性反馈移位寄存器单元与输出端口连接;设置第四开关,以与线性反馈移位寄存器单元的反馈回路断开连接。可使第二线性反馈移位寄存器单元移位L次,以获得用于码块的循环冗余校验。

[0032] 根据本发明的另一方面,提供一种在数据通信中生成循环冗余校验的电路,所述电路包括:输入端口,用于接收信息数据;输出端口,用于输出信息数据和循环冗余校验;线性反馈移位寄存器单元,在输入端口和输出端口之间通信连接,包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据;第一开关,在输入端口和线性反馈移位寄存器单元之间通信连接;第二开关,通信连接在线性反馈移位寄存器单元的反馈回路;第三开关,在输入端口、线性反馈移位寄存器单元和输出端口之间通信连接,具有将输入端口与输出端口连接的第一位置以及使线性反馈移位寄存器单元与输出端口连接的第二位置;

[0033] 可将线性反馈移位寄存器单元初始化为全零状态。可设置第一开关以将输入端口连接到线性反馈移位寄存器单元;可设置第二开关以连接线性反馈移位寄存器单元的反馈回路;可将第三开关设置到第一位置,以使输入端口与输出端口连接。可经输入端口接收信息数据的码块。可设置第一开关以使输入端口与线性反馈移位寄存器单元断开连接;可设置第二开关以断开与线性反馈移位寄存器单元的反馈回路的连接;可将第三开关设置到第二位置,以使线性反馈移位寄存器单元与输出端口连接。可使线性反馈移位寄存器单元移位L次,以获得用于码块的循环冗余校验。

附图说明

[0034] 通过参考下面结合附图的详细描述,本发明的更全面的理解及其许多优点将更容易理解,从而将变得明显,在附图中,相同的标号表示相同或相似的部件,其中:

[0035] 图1是适合于本发明原理的实践的正交频分复用(OFDM)收发机链的示图;

[0036] 图2是示出作为频率的函数的振幅的OFDM子载波的两个坐标曲线图;

[0037] 图3是时域中的OFDM符号的波形图;

[0038] 图4是单载波频分多址收发机链的示图;

[0039] 图5示意性地示出混合自动重传请求(HARQ)收发机链;

[0040] 图6示意性地示出多入多出(MIMO)系统;

[0041] 图7示意性地示出预编码的MIMO系统;

[0042] 图8示意性地示出高速下行链路分组接入(HSDPA)系统中的高速数据共享信道(HS-DSCH)的编码链;

[0043] 图9示意性地示出传输块循环冗余校验(CRC)和码块分割;

[0044] 图10是使用线性反馈移位寄存器(LFSR)用于CRC计算的示图;

[0045] 图11示意性地示出高速数据共享信道(HS-DSCH)混合ARQ功能;

- [0046] 图12示意性地示出长期演进(LTE)下行链路子帧结构;
- [0047] 图13示意性地示出LTE上行链路子帧结构;
- [0048] 图14示意性地示出码块CRC;
- [0049] 图15示出码块分割的示例;
- [0050] 图16示意性地示出根据本发明原理的一个实施例的码块(CB)CRC和传输块(TB)CRC;
- [0051] 图17示意性地示出根据本发明原理的另一实施例的码块(CB)CRC和传输块(TB)CRC;
- [0052] 图18示意性地示出根据本发明原理的另一实施例的码块(CB)CRC和传输块(TB)CRC;
- [0053] 图19示意性地示出根据本发明原理的另一实施例的码块(CB)CRC和传输块(TB)CRC;
- [0054] 图20示意性地示出根据本发明原理的另一实施例的码块(CB)CRC和传输块(TB)CRC;
- [0055] 图21示意性地示出构造为本发明原理的实施例的用于多个码块的CRC计算电路;
- [0056] 图22示意性地示出构造为本发明原理的另一实施例的用于多个码块的CRC计算电路;
- [0057] 图23示意性地示出构造为本发明原理的另一实施例的用于多个码块的CRC计算电路;和
- [0058] 图24示意性地示出构造为本发明原理的另一实施例的用于多个码块的CRC计算电路。

具体实施方式

[0059] 正交频分复用(OFDM)是在频域中复用数据的技术。在频率子载波上携带调制符号。图1示出正交频分复用(OFDM)收发机链。在使用OFDM技术的通信系统中,在发送机链110,由调制器112将控制信号或数据111调制为一串调制符号,随后由串/并(S/P)转换器113对这些调制符号进行串行至并行的转换。逆快速傅立叶变换(IFFT)单元114用于将信号从频域转移到时域,成为多个OFDM符号。由循环前缀(CP)插入单元116将CP或零前缀(ZP)添加到每一OFDM符号,以避免或减轻由多径衰落带来的影响。因此,通过发送机(Tx)前端处理单元117,如天线(未示出),或者可选地,通过固定的电线或电缆来发送信号。在接收机链120,假设实现了完全的时间和频率同步,则通过接收机(Rx)前端处理单元121接收的信号被CP去除单元122处理。快速傅立叶变换(FFT)单元124将接收的信号从时域转移到频域,以便进一步处理。

[0060] 在OFDM系统中,每一OFDM符号由多个子载波构成。OFDM符号内的每一子载波携带调制符号。图2示出使用子载波1、子载波2和子载波3的OFDM传输方案。由于每一OFDM符号在时域中具有有限的持续时间,所以子载波在频域中彼此重叠。然而,如图2所示,假设发送机和接收机具有完全的频率同步,则在采样频率上保持正交性。在由于不完全频率同步或者高移动性引起的频率偏移的情况下,在采样频率上子载波的正交性被破坏,导致载波间干扰(ICI)。

[0061] 图3中示出了发送和接收的OFDM符号的时域示意图。如图3所示,由于多径衰落,接收的信号CP部分(CP1、CP2)经常被先前的OFDM符号所破坏。然而,只要CP足够长,接收的没有CP的OFDM符号就应该仅包含其自身的通过多径衰落信道卷积的信号。通常,在接收机方采取快速傅立叶变换(FFT)以允许进一步处理频域。OFDM较其它传输方案的优势是其对多径衰落的鲁棒性。时域中的多径衰落转化为频域中的频率选择性衰落。通过添加的循环前缀或者零前缀,避免或者极大缓和了相邻OFDM之间的符号间干扰。另外,由于在窄带宽上运送每一调制符号,所以其经受单径衰落。可使用简单的均衡方案来防止频率选择性衰落。

[0062] 采用单载波调制和频域均衡的单载波频分多址(SC-FDMA)是一种性能和复杂度与OFDMA系统相似的技术。SC-FDMA的一个优势在于:由于SC-FDMA所固有的单载波结构,所以SC-FDMA信号具有较低的峰均功率比(PAPR)。通常PAPR低导致功率放大器的效率高,这对上行链路传输中的移动站特别重要。在3GPP长期演进(LTE)中选择SC-FDMA作为上行链路多路接入方案。图4示出了SC-FDMA的收发机链的例子。在发送机方,由S/P转换器401对数据或控制信号进行串行至并行转换。在由子载波映射单元403将时域数据映射到一组子载波之前,将由离散傅立叶变换(DFT)变换器402将DFT应用到时域数据或控制信号。为了确保低PAPR,通常,频域中的DFT输出将被映射到一组邻接的子载波。然后,将由IFFT变换器404应用大小通常大于DFT的IFFT,以将信号变换回时域。在通过P/S转换器405进行并行至串行(P/S)转换之后,并且在数据或控制信号被发送到发送机前端处理单元407之前,将由循环前缀(CP)插入单元406将CP添加到数据或控制信号。添加了循环前缀的经处理的信号经常被称为SC-FDMA块。在信号经过无线通信系统中的通信信道408,例如多径衰落信道之后,接收机将通过接收机前端处理单元409执行接收机前端处理,通过CP去除单元410去除CP,通过FFT变换器412应用FFT,并进行频域均衡413。在均衡的信号在频域中被解映射之后,将应用逆离散傅立叶变换(IDFT)414。在通过并串(P/S)转换器415对IDFT的输出进行P/S转换之后,所述输出将被传递以进行进一步的时域处理,如解调和解码。

[0063] 在基于包的无线数据通信系统中,通过控制信道传输的控制信号(即,控制信道传输)通常伴随着通过数据信道传输的数据信号(即,数据传输)。包括控制信道格式指示符(CCFI)、应答信号(ACK)、分组数据控制信道(PDCCH)信号的控制信道信息携带用于数据信号的传输格式信息,如用户ID、资源分配信息、有效载荷(Payload)大小、调制、混合自动重传请求(HARQ)信息、MIMO相关信息。

[0064] 混合自动重传请求(HARQ)广泛用在通信系统中,以防止解码失败并提高可靠性。图5示意性地示出一般的包括编码器501、子包(subpacket)生成器502、收发机链503和解码器504的混合自动重传请求(HARQ)收发机链。利用特定的前向纠错(FEC)方案对每一数据包进行编码。在子包生成器502中生成的每一子包可仅包含编码的位(bit)的一部分。如果子包k的传输失败,如反馈应答信道505中的NAK所指示,则重传子包(子包k+1)被发送,以帮助接收机对包进行解码。重传子包可包含不同于先前子包的编码的位。接收机可适当地将所有接收的子包结合,或者对所有接收的子包进行联合解码,以提高解码的可能性。通常,考虑可靠性、包延迟以及实现复杂度来配置传输的最大数量。

[0065] 经常被称为多入多出(MIMO)的多天线通信系统广泛用在无线通信中,以提高系统性能。在如图6所示的MIMO系统中,发送机601具有能够发送独立的信号的多个天线602,接收机603配备有多个接收天线604。如果仅有一个发送天线,或者如果仅有一个发送的数据

流,则MIMO系统退化为单入多出(SIMO)。如果仅有一个接收天线,则MIMO系统退化为多入单出(MISO)。如果仅有一个发送天线和一个接收天线,则MIMO系统退化为单入单出(SISO)。MIMO技术可显著提高系统的吞吐量和范围,而不增加带宽或总传输功率使用。通常,由于有多个天线,MIMO技术通过开发空间域中的额外的自由度来提高无线通信系统的频谱效率。有多种MIMO技术。例如,空间复用方案通过允许经多个天线传输多个数据流来增加传输速率。诸如空时编码的发送多样性方法利用了由于多个发送天线引起的空间多样性。接收机多样性方法利用由于多个接收天线引起的空间多样性。波束形成技术提高了接收的信号增益并减小对其它用户的干扰。空分多址(SDMA)允许来自多个用户或者至多个用户的信号流经相同的时间-频率资源被传输。接收机可通过多个数据流的空间特征来分离这些数据流。需要注意的是,这些MIMO传输技术不是互斥的。实际上,许多MIMO方案经常用在高级无线系统中。

[0066] 当信道较有利,例如,移动速度低时,可使用闭环MIMO方案,以改善系统性能。在闭环MIMO系统中,接收机反馈信道条件和/或优选的Tx MIMO处理方案。发送机利用该反馈信息,连同其它考虑(如调度优先级、数据和资源可用性)一起,对发送方案一同进行优化。流行的闭环MIMO方案被称为MIMO预编码。通过预编码,在发送数据流被传送到多个发送天线之前,将发送数据流与矩阵预先相乘。如图7所示,假设有 N_t 个发送天线702和 N_r 个接收天线704。将 N_t 个发送天线702与 N_r 个接收天线704之间的信道表示为 H 。因此, H 是 $N_t \times N_r$ 矩阵。如果发送机701知道 H ,则发送机可根据 H 选择最有利的发送方案。例如,如果吞吐量最大化是目标,则如果发送机已知 H ,可选择预编码矩阵为 H 的右奇异矩阵(right singular matrix)。通过这样做,可将接收机方703多个数据流的有效信道对角线化,消除了多个数据流之间的干扰。然而,经常不容许有反馈 H 的准确值所需的开销。为了减少反馈开销,定义一组预编码矩阵,以对 H 可具有的可能值的空间进行量化。通过量化,接收机通常以优选预编码矩阵的索引、秩(rank)以及优选预编码矢量的索引的形式反馈优选预编码方案。接收机还可反馈优选预编码方案的相关CQI值。

[0067] MIMO系统的另一方面是:用于传输的多数据流是被分离地编码还是被一起编码。如果所有用于传输的层被一起编码,则我们称其为单码字(SCW)MIMO系统。

[0068] 在LTE系统中,当传输块较大时,传输块被分割为多个码块,从而可生成多个编码的包,由于诸如能够并行处理或者流水线执行以及功耗和硬件复杂度之间的柔性权衡的好处,这样做有利。作为示例,图8示出了高速下行分组接入(HSDPA)系统中的高速数据共享信道(HS-DSCH)的编码处理。在当前的HS-DSCH设计中,为整个传输块仅生成一个24位的循环冗余校验(CRC),以便对该块进行错误检测。如果生成多个码块并且在一个传输时间间隔(TTI)中发送所述多个码块,则接收机可能正确地将一些码块解码,而不能正确地将其它码块解码。在这种情况下,由于用于传输块的CRC将证明有误,所以接收机将向发送机反馈非应答(NAK)。在图9中,标号901至905示出传输块、传输块CRC(TB CRC)以及码块分割之间的关系。

[0069] 假设我们使用 L 位CRC多项式来生成码块CRC。CRC生成多项式被表示为:

$$g(x) = g_0x^L + g_1x^{L-1} + \dots + g_{L-1}x + g_L. \quad (1)$$

[0071] 通常,对于消息:

$$m(x) = m_0x^{M-1} + m_1x^{M-2} + \dots + m_{M-2}x + m_{M-1}, \quad (2)$$

[0073] 以系统形式执行CRC编码。将消息的CRC奇偶校验位表示为 $p_0, p_1, \dots, p_{L-1}$, 这也可被表示为多项式:

$$p(x) = p_0x^{L-1} + p_1x^{L-2} + \dots + p_{L-2}x + p_{L-1}. \quad (3)$$

[0075] 当多项式

$$m(x) \cdot x^L - p(x) = m_0x^{M+L-1} + m_1x^{M+L-2} + \dots + m_{M-2}x^{L+1} + m_{M-1}x^L + p_0x^{L-1} + p_1x^{L-2} + \dots + p_{L-2}x + p_{L-1} \quad (4)$$

[0077] 被 $g(x)$ 除时, 产生等于0的余数。

[0078] 需要注意的是, 如果该消息中的每一位是二进制的, 则可将该消息表示为二进制伽罗瓦域(GF(2))的多项式。在这种情况下, “+”和“-”运算是相同的。换句话说, 如果消息位是二进制的, 则可通过 $m(x) \cdot x^L + p(x)$ 或者 $m(x) \cdot x^L - p(x)$ 来表示附着有CRC的消息。在本发明的剩余部分中, 为了方便起见, 我们假设消息位是二进制的。然而, 当消息位是非二进制时, 本发明中公开的构思当然也可适用。

[0079] CRC流行的一个原因在于其实现的简单性。可通过线性反馈移位寄存器(LFSR)来容易地实现CRC计算。可使用LFSR作为用于多项式除法的电路。如图10所示, 假设使用 L 位CRC, LFSR1000具有 L 个移位寄存器($R_0 \sim R_{L-1}$)。开关1001、1003和1005初始被置于位置X。按照索引增大的次序, 将消息位 $m_0, m_1, \dots, m_{M-1}$ 一次一个地供应给LFSR1000。在将最后一位(m_{M-1})供应给LFSR1000之后, 开关1001、1003和1005移到位置Y。使LFSR1000另外移位 L 次, 以在最右侧的寄存器的输出端产生CRC。需要注意的是, 图9的LFSR仅是示例。当然存在用于多项式除法和CRC计算的LFSR的其它实现方式。

[0080] 混合ARQ功能将信道编码器的输出端的位的数量与高速数据共享信道(HS-DSCH)所映射到的高速物理下行链路共享信道(HS-PDSCH)集合的位的总数相匹配。混合ARQ功能受冗余版本(RV)参数控制。混合ARQ功能的输出端的位的准确集合依赖于输入位的数量、输出位的数量以及RV参数。如图11所示, 混合ARQ功能包括两个速率匹配级1101和1103以及虚拟缓冲器1105。位分离级1107将信道编码器的输出位分割为系统位、奇偶校验位1和奇偶校验位2, 并将其输入到速率匹配级。第一速率匹配级1101将输入位的数量匹配到虚拟IR缓冲器1105, 其中, 关于虚拟IR缓冲器1105的信息由更高层提供。需要注意的是, 如果输入位的数量不超过虚拟IR缓冲能力, 则第一速率匹配级1101是透明的。第二速率匹配级1103将第一速率匹配级1101的输出端的位的数量与TTI中的HS-PDSCH集合中可用的物理信道位的数量相匹配。位收集级1109收集第二速率匹配级1103的输出位, 并将其发送给无线网络。

[0081] 图12示出了LTE的下行链路子帧结构。在典型配置中, 如垂直轴所示, 每一子帧为1ms长, 包含14个OFDM符号。假设对子帧中的OFDM符号从0至13编索引。用于天线0和1的参考符号(RS)位于OFDM符号0(1201)、4(1203)、7(1205)和11(1207)中。如果存在, 用于天线2和3的参考符号(RS)位于OFDM符号1(1211)和8(1213)中。在开始的一个、或者两个、或者三个OFDM符号中发送包括控制信道格式标识符(CCFI)、应答信道(ACK)、分组数据控制信道(PDCCH)的控制信道。由CCFI来指示用于控制信道的OFDM符号的数量。例如, 控制信道可占据开始的一个OFDM符号, 或者开始的两个OFDM符号, 或者开始的三个OFDM符号。在其它OFDM符号中发送数据信道, 即, 物理下行链路共享信道(PDSCH)。

[0082] 图13示出了上行链路子帧结构(用于数据传输)。需要注意的是, LTE上行链路是基于SC-FDMA的系统, 其与OFDMA系统很相像, 仅有一些不同。与OFDM符号类似, 每一SC-FDMA块具有循环前缀(CP)。为了数据传输, 参考信号(RS)位于第4个SC-FDMA块1301和第11个SC-

FDMA块1303中,而剩余SC-FDMA块携带数据。需要注意的是,图13仅示出了上行链路子帧的时域结构。对于每一单独的UE,其传输可仅占据频域中的整个带宽的一部分。在频域中经SC-FDMA复用不同的用户和控制信号。

[0083] 在本发明中,我们提出这样的方法和设备,所述方法和设备计算用于传输的多个CRC,以提高传输的可靠性并降低发送机和接收机的复杂度。

[0084] 通过仅示出若干特定实施例和实施方式(包括实现本发明所预期的最佳方式)从下面详细的描述中,本发明的各方面、特点和优点将明显。本发明还能够有其它和不同的实施例,在不脱离本发明的精神和范围的情况下,可对其若干细节在各个明显方面进行修改。因此,附图和说明书意在被认为本质上是说明性的,而非限制性的。在附图中,本发明通过示例的方式,而非限制的方式被说明。在下面的说明中我们使用LTE系统中的数据信道作为示例。然而,只要可适用,这里所说明的技术当然可用在LTE系统的其它信道中、以及其它系统中的其它数据信道、控制信道或其它信道中。

[0085] 我们首先说明传输块、码块和码块循环冗余校验(CRC)的概念。图14示出了发送机方的编码处理链的一部分。如果需要,可将TTI中的多个传输块串接。如果在传输块串接(concatenation)之后,位的数量大于Z(Z是所涉及的码块的最大大小),则在传输块的串接之后执行码块分割。需要注意的是,在本发明中,在分割之前,传输块可以包含传输块CRC,或者可以不包含传输块CRC。在码块分割之后,可针对一些码块或者所有码块生成CRC。在码块CRC被附着到相应码块之后,信道编码器对附着有CRC的码块进行编码。混合ARQ功能使从信道编码器输出的位的数量与高速数据共享信道(HS-DSCH)所映射到的高速物理下行链路共享信道(HS-PDSCH)集合的位的总数相匹配。为了说明的目的,尽管本发明中所公开的构思当然可以以其它方式应用,但假设为每一码块生成码块CRC。为了说明简单起见,我们假设仅存在一个传输块。然而,本发明中的所有实施例均可应用于具有多个传输块且传输块串接的情况。另外,需要注意的是,尽管我们经常使用发送机处理来说明本发明的构思,但是本发明的所有实施例均可应用于发送机和接收机二者的CRC计算。

[0086] 通过 $a_0, a_1, \dots, a_{A-1}$ 来表示CRC计算的输入位,其中,A是传输块的大小。我们称输入位为信息位。此外,无论是存在一个还是多个传输块,或者传输块是否包含传输块CRC,本发明中所描述的方法均适用。假设我们使用L位CRC多项式来生成码块CRC。将CRC生成多项式表示为:

$$[0087] \quad g(x) = g_0x^L + g_1x^{L-1} + \dots + g_{L-1}x + g_L. \quad (5)$$

[0088] 传输块可由下面的多项式表示:

$$[0089] \quad a(x) = a_0x^{A-1} + a_1x^{A-2} + \dots + a_{A-2}x + a_{A-1}. \quad (6)$$

[0090] 假设通过码块分割,传输块消息被分割为C个码块。通过 $b_{i,0}, b_{i,1}, \dots, b_{i,(B_i-1)}$ 来表示码块i中的位,其中, B_i 是码块i的大小, $i=0, 1, \dots, C-1$ 。需要注意的是,码块i的多项式表达为:

$$[0091] \quad b_i(x) = b_{i,0}x^{B_i-1} + b_{i,1}x^{B_i-2} + \dots + b_{i,(B_i-2)}x + b_{i,(B_i-1)}. \quad (7)$$

$$[0092] \quad \text{很明显, } A = \sum_{i=0}^{C-1} B_i.$$

[0093] 在不损失一般性的情况下,如图15所示,我们进一步假设从传输块1501中的信息

位映射到码块1503、……、1505的自然次序。我们定义：

$$[0094] \quad D_i = \sum_{j=0}^{i-1} B_j, \text{ 对于 } i=0, 1, \dots, C. \quad (8)$$

$$[0095] \quad \text{换言之, } D_0=0, D_1=B_0, D_2=B_0+B_1, \dots, D_C = \sum_{i=0}^{C-1} B_i = A.$$

[0096] 传输块被分割为C个码块。第i码块的信息位如下所示：

$$[0097] \quad b_{i,0} = a_{D_i}, b_{i,1} = a_{D_i+1}, \dots, b_{i,B_i} = a_{D_{i+1}-1}, \text{ 对于 } i=0, 1, \dots, C-1. \quad (9)$$

[0098] 以这样的方式,传输块与码块之间的关系可建立为如下所示：

$$[0099] \quad a(x) = \sum_{i=0}^{C-1} [b_i(x) \cdot x^{A-D_{i+1}}]. \quad (10)$$

[0100] 我们进一步定义：

$$[0101] \quad a_k(x) = \sum_{i=0}^k [b_i(x) \cdot x^{D_{k+1}-D_{i+1}}], \text{ 对于 } k=0, 1, \dots, C-1. \quad (11)$$

[0102] 其中, $a_k(x)$ 是直到第k码块的信息位(包括先前码块中的信息位)的多项式表达。很容易看出, $a_0(x) = b_0(x)$, $a_{C-1}(x) = a(x)$ 。为了简单起见,在本发明的剩余部分中仍使用这些符号,而不再进行重复的定义。

[0103] 在根据本发明原理的第一实施例中,在具有第二数量的多个CRC的第一数量的多个位的发送过程中或者在这样的发送的接收处理过程中,基于所述第一数量的多个位的位子集计算至少一个CRC,使得所述第一数量的多个位中的至少一位不在所述子集中。在图16所示的示例中,从传输块生成传输块CRC(TB CRC),包括传输块CRC的传输块1601被分割为码块01603、码块11605、码块21607。基于码块01603中的信息位,但是不基于码块11605或者码块21607中的信息位,来计算CB0_CRC1609。在这样做时,在完成对码块11605和码块21607的接收机处理之前,UE可使用CB0_CRC1609来校验码块01603中的信息位是否被正确接收。这一特点在降低UE复杂度和节省功率方面特别有益。可出于例如(但不限于)这样的目的来使用码块CRC:为相应一个或多个码块提供错误检测,提早停止迭代turbo解码从而可实现功率节省以及码块之间的解码能力的统计复用,检测一个码块的解码错误从而能够避免在一个码块解码错误的情况下对其它码块进行不必要的解码等。

[0104] 在根据本发明原理的第二实施例中,在具有第二数量的多个CRC的第一数量的多个位的发送过程中或者在这样的发送的接收机处理过程中,基于所述第一数量的多个位子集计算至少一个CRC,其中,通过某一类型的前向纠错码对所述子集进行联合编码。例如,如图16所示,基于传输块中的所有位子集(码块01603中的位)来计算CB0_CRC1609。通过某一前向纠错(FEC)码,如turbo码来对码块01603中的位进行联合编码。FEC编码有时也被称为信道编码。需要注意的是,通常也将CB0_CRC1609与码块01603中的信息位联合编码,以实现信息位和CRC位二者的错误保护。通过将用于CRC计算的信息位的块以及用于FEC信道编码的信息位的块进行同步,UE可在解码处理以及确定相应码块是否被正确解码期间使用码块CRC。可以以并行或流水线以及串行方式,用码块CRC对每一码块分别进行这一处理。

[0105] 在根据本发明原理的第三实施例中,在具有第二数量的多个CRC的第一数量的多个位的发送过程中或者在这样的发送的接收处理过程中,基于位的第一子集计算第一CRC,

同时基于位的第二子集计算第二CRC。图16示出了一个示例。在该示例中，“位子集”被称为码块。为传输块计算包括传输块CRC的传输块1601。然后，传输块被分割为三个码块。为每一码块计算CRC。CB0_CRC1609是附着到码块01603的码块CRC，基于码块0中的位来得到CB0_CRC1609；CB1_CRC1611是附加到码块11605的码块CRC，基于码块1中的位来得到CB1_CRC1611；CB2_CRC1613是附加到码块21607的码块CRC，基于包括传输块CRC的码块2中的位来得到CB2_CRC1613。还需要注意的是，在本示例中，用于得到第一CRC的位的第一子集不与用于得到第二CRC的位的第二子集重叠。然而，在不脱离本发明的公开的情况下，位的字节当然也可以重叠。还需要注意的是，一些子集可包括传输中的所有位。还需要注意的是，不是必需为所有码块计算CRC，以便使用本发明。一些码块可以不具有码块CRC。还需要注意的是，子集还可包括多个码块中的位。例如，如图17所示，基于包括码块01703中的位的位子集来得到CB0_CRC1709；基于包括码块01709中的位和码块11705中的位的位子集来得到CB1_CRC1711；基于包括码块01703中的位、码块11705中的位和码块21707中的位的位子集来得到CB2_CRC1713。

[0106] 在根据本发明原理的第四实施例中，在具有第二数量的多个CRC的第一数量的多个位的发送过程中或者在这样的发送的接收处理过程中，用于得到第一CRC的位是用于得到第二CRC的位子集。图17示出了一个示例。为了说明的目的，我们仅示出了三个码块。为传输块计算传输块CRC。然后，包括传输块CRC的传输块1701被分割为三个码块。为每一码块计算CRC。CB0_CRC1709是附着到码块01703的码块CRC，基于码块0中的位来得到CB0_CRC1709；CB1_CRC1711是附加到码块11705的码块CRC，基于码块0和码块1中的位来得到CB1_CRC1711；CB2_CRC1713是附加到码块21707的码块CRC，基于码块01703、码块11705和码块21707中的位来得到CB2_CRC1713。这样，与基于单个码块得到的CRC相比，我们改善了这些CRC的丢失检测性能。我们假设传输块为 $a(x) = a_0x^{A-1} + a_1x^{A-2} + \dots + a_{A-2}x + a_{A-1}$ ，其中，A是传输块大小。如果使用传输块CRC(TB_CRC)，则TB_CRC被包括在消息中。如前面所定义的，传输块a(x)被分割为C个码块，码块i由 $b_i(x)$ 表示。我们计算一个CRC，即CB0_CRC，并将其附着到第一码块。CB0_CRC可得自第一码块中的一些位或者所有位。我们将CB0_CRC表示为：

$$[0107] \quad p_0(x) = p_{0,0}x^{L-1} + p_{0,1}x^{L-2} + \dots + p_{0,L-2}x + p_{0,L-1} \quad (12)$$

[0108] 计算CB0_CRC的一个例子是：求 $b_0(x) \cdot x^L$ 被CRC生成器多项式 $g(x)$ 除所得的余数，其中， $p_0(x)$ 可被表示为：

$$[0109] \quad p_0(x) = b_0(x) \cdot x^L - q_0(x) \cdot g(x) \quad (13)$$

[0110] 其中， $q_0(x)$ 是 $b_0(x) \cdot x^L$ 被 $g(x)$ 除所得的商。我们计算另一CRC，即CB1_CRC，并将其附着到第二码块。CB1_CRC可得自第一码块中的一些位或所有位以及第二码块中的一些位或所有位。我们将CB1_CRC表示为：

$$[0111] \quad p_1(x) = p_{1,0}x^{L-1} + p_{1,1}x^{L-2} + \dots + p_{1,L-2}x + p_{1,L-1} \quad (14)$$

[0112] 计算CB1_CRC的一个例子是：求 $(b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L$ 被CRC生成器多项式 $g(x)$ 除所得的余数，其中， $p_1(x)$ 可被表示为：

$$[0113] \quad p_1(x) = (b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L - q_1(x) \cdot g(x), \quad (15)$$

[0114] 其中， $q_1(x)$ 是 $(b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L$ 被 $g(x)$ 除所得的商。通过基于第一码块中的信息位和第二码块中的信息位得到CB1_CRC，我们减小了丢失检测概率，这是因为可使用CB1_

CRC检测第一码块和第二码块中的信息位的错误。

[0115] 很明显,如果存在两个以上的码块,我们可以以相似的方式延伸操作。例如,附着到码块2的CRC可得自码块0、码块1和码块2中的位。可选地,附着到一个码块的CRC不需要基于包括当前码块的所有先前码块中的位来获得。例如,附着到码块2的CRC可得自码块1和码块2中的位,但不得自码块0中的位。需要注意的是,如图18所示,本实施例还应用在没有传输块CRC时。如果CB_CRC的错误检测足够可靠,则可不需要TB_CRC。

[0116] 在根据本发明原理的第五实施例中,在具有第二数量的多个CRC的第一数量的多个位的发送过程中或者在这样的发送的接收处理过程中,在码块分割之前,从传输块的所有位得到传输块CRC,同时存在没有从其计算码块CRC的至少一个位子集。如图19所示,基于传输块中的位计算传输块CRC。然后,包括传输块CRC的传输块1901被分割为三个码块。在本示例中,基于码块01903中的位计算CB0_CRC1909;基于码块01903和码块11905中的位计算CB1_CRC1911。然而,由于存在覆盖传输块中的所有位的传输块CRC,所以码块2的码块CRC不是必需的。CB0_CRC1909可用于停止对码块01903的turbo解码迭代;CB1_CRC1911可用于停止对码块11905的turbo解码迭代;TB_CRC可用于停止对码块21907的turbo解码迭代。此时,TB_CRC提供对整个传输块的错误检测。

[0117] 在根据本发明原理的第六实施例中,在具有第二数量的多个CRC的第一数量的多个信息位的发送过程中或者在这样的发送的接收处理过程中,从所有信息位得到第一CRC,同时从信息位的子集得到第二CRC。如图20所示,在码块分割之前,没有计算传输块CRC。传输块2001被分割为三个码块。为三个码块中的每一码块计算码块CRC。从码块02003中的位得到CB0_CRC2009;从码块12005中的位得到CB1_CRC2011;从码块02003、码块12005和码块22007中的位得到CB2_CRC2013。CB0_CRC2009可用于停止对码块02003的turbo解码迭代或错误检测;CB1_CRC2011可用于停止对码块12005的turbo解码迭代或错误检测;CB2_CRC2013可用于停止对码块22007的turbo解码迭代和错误检测。此时,CB2_CRC2013提供对整个传输块的错误检测。

[0118] 在下面的实施例中,我们说明可如何使用基于线性反馈移位寄存器(LFSR)的电路有效地为多个信息位计算多个CRC。需要注意的是,尽管为了说明的目的,我们使用了发送机方CRC生成,但是对于本领域普通技术人员而言简单的是,将这些方法应用于接收机处理。为了方便起见,我们假设CRC计算电路被初始化为全零状态。然而,当LFSR被设置为非零状态时,本发明中所公开的构思也可应用。

[0119] 在根据本发明原理的第七实施例中,可以用一单个CRC计算电路递归地计算用于第二数量的多个信息位的第一数量的多个CRC。我们计算一个CRC,即CB0_CRC,并将其附着到第一码块。CB0_CRC可得自第一码块中的一些位或者所有位。我们将CB0_CRC表示为:

$$[0120] \quad p_0(x) = p_{0,0}x^{L-1} + p_{0,1}x^{L-2} + \dots + p_{0,L-2}x + p_{0,L-1}. \quad (16)$$

[0121] 计算CB0_CRC的一个例子是:求 $b_0(x) \cdot x^L$ 被CRC生成器多项式 $g(x)$ 除所得的余数,其中, $p_0(x)$ 可被表示为:

$$[0122] \quad p_0(x) = b_0(x) \cdot x^L - q_0(x) \cdot g(x) \quad (17)$$

[0123] 其中, $q_0(x)$ 是 $b_0(x) \cdot x^L$ 被 $g(x)$ 除所得的商。我们计算另一CRC,即CB1_CRC,并将其附着到第二码块。CB1_CRC可得自第一码块中的位以及第二码块中的位。我们将CB1_CRC表示为:

$$[0124] \quad p_1(x) = p_{1,0}x^{L-1} + p_{1,1}x^{L-2} + \dots + p_{1,L-2}x + p_{1,L-1}. \quad (18)$$

[0125] 换句话说, CB1_CRC 是 $(b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L$ 被 CRC 生成器多项式 $g(x)$ 除所得的余数, 其中, $p_1(x)$ 可被表示为:

$$[0126] \quad p_1(x) = (b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L - q_1(x) \cdot g(x), \quad (19)$$

[0127] 其中, $q_1(x)$ 是 $(b_0(x) \cdot x^{B_1} + b_1(x)) \cdot x^L$ 被 $g(x)$ 除所得的商。通过基于第一码块中的信息位和第二码块中的信息位来得到 CB1_CRC, 我们减小了丢失检测概率, 这是因为可使用 CB1_CRC 检测第一码块和第二码块中的错误。类似地, 可将用于第 k 码块的 CRC 计算为 $a_k(x) \cdot x^L$ 被 $g(x)$ 除所得的余数。换句话说,

$$[0128] \quad p_k(x) = a_k(x) \cdot x^L - q_k(x) \cdot g(x) \quad (20)$$

[0129] 其中, $q_k(x)$ 是 $a_k(x) \cdot x^L$ 被 $g(x)$ 除所得的商。需要注意的是:

$$[0130] \quad a_k(x) = a_{k-1}(x) \cdot x^{B_k} + b_k(x). \quad (21)$$

[0131] 这种计算 CRC 的方式使其自身成为简单的 CRC 计算方法。用于第 k 码块的 CRC 可被表示为:

$$[0132] \quad \begin{aligned} p_k(x) &= [a_{k-1}(x) \cdot x^L \cdot x^{B_k} + b_k(x) \cdot x^L] - q_k(x) \cdot g(x) \\ &= p_{k-1}(x) \cdot x^{B_k} + b_k(x) \cdot x^L + [q_{k-1}(x) \cdot x^{B_k} - q_k(x)] \cdot g(x). \end{aligned} \quad (22)$$

[0133] 换言之, $a_k(x)$ 的 CRC 与 $p_{k-1}(x) \cdot x^{B_k} + b_k(x) \cdot x^L$ 的 CRC 相同。

[0134] 图 21 中示出了递归地计算 CRC $p_i(x)$ 的电路的一个示例, 其中, $i=0, 1, \dots, C-1$ 。如图 21 中所示, 用输入端口 2109、输出端口 2111 和线性反馈移位寄存器 (LFSR) 单元 2100 来构造用于计算 CRC 的电路, 其中, 输入端口 2109 用于接收信息数据, 输出端口 2111 用于输出信息数据和 CRC, LFSR 单元 2100 在输入端口 2109 和输出端口 2111 之间通信连接。LFSR 单元 2100 包括 L 个移位寄存器 2115、 L 个与门 2113 (用环绕圆圈的 “ $\times$ ” 表示) 和 L 个异或门 2117 (用环绕圆圈的 “ $+$ ” 表示)。循环冗余校验寄存器单元 2119 连接在输入端口 2109 和 LFSR 单元 2100 之间。循环冗余校验寄存器单元 2119 包括 L 个循环冗余校验寄存器。第一开关 2101 位于输入端口 2109 和循环冗余校验寄存器单元 2119 之间。第一开关具有位置 X 和位置 Y, 其中, 位置 X 使输入端口 2109 和循环冗余校验寄存器单元 2119 连接, 位置 Y 使输入端口 2109 和循环冗余校验寄存器单元 2119 之间断开连接。第二开关 2103 位于 LFSR 单元 2100 的反馈回路。第二开关 2103 具有连接 LFSR 单元 2100 的反馈回路的位置 X 和与 LFSR 单元 2100 的反馈回路断开连接的位置 Y。第三开关 2105 位于 LFSR 单元 2100 和循环冗余校验寄存器单元 2119 之间。第三开关 2105 具有使 LFSR 单元 2100 与循环冗余校验寄存器单元 2119 断开连接的位置 X 以及使 LFSR 单元 2100 与循环冗余校验寄存器单元 2119 连接的位置 Y。第四开关 2107 位于输入端口 2109、LFSR 单元 2100 和输出端口 2111 之间。第四开关 2107 具有使输入端口 2109 与输出端口 2111 连接的位置 X 以及使 LFSR 单元 2100 与输出端口 2111 连接的位置 Y。所述开关可以是当前的任何一种电子开关, 例如当前的任何一种场效应晶体管 (FET)。操作图 21 所示的电路的相应过程概述如下:

[0135] 将 LFSR 单元 2100 初始化为全零状态; 设置 $k=0$; 将 CRC 寄存器初始化为零; 将所有开关 2101、2103、2105 和 2107 均设置在位置 X。

[0136] 将 $b_k(x)$ 一次一位地输入到电路中。需要注意的是, 对于每一位输入, LFSR 也移位

一次。

[0137] 将所有开关2101、2103、2105和2107均改变到位置Y。

[0138] 使LFSR单元2100和CRC寄存器移位L次,以输出作为L位CRC的 $p_k(x)$ 。

[0139] 将LFSR单元2100重置为全零状态;将所有开关改换到位置X。

[0140] $k+1$ 。

[0141] 如果 $k < C$,则转到步骤2。

[0142] 附着到第k码块的CRC可由等式20表示: $p_k(x) = a_k(x) \cdot x^L - q_k(x) \cdot g(x)$ 。换句话说,基于第k码块以及先前所有码块的信息位来计算第k码块的CRC。如我们所看到的,除了在将每一码块输入到电路之后,CRC应该被存储并在某个时候被添加回以用于下一码块的CRC计算之外,码块CRC的计算与传输块CRC的计算相同。以这样的方式,避免了分别用于计算码块CRC和传输块CRC的单独的电路和额外的计算复杂度。实际上,最后一个码块CRC等于传输块CRC。这种结构非常适合多码块的流水线结构。另外,至少可确保传输块的丢失检测性能。需要注意的是,等式(20)仅是针对基于第k码块和所有先前的码块的信息位计算第k码块CRC时的情形。

[0143] 可选地,根据本发明原理的第八实施例,图22中示出了递归地计算 $CRC_{pi}(x)$ 的电路的另一个示例,其中, $i=0,1,\dots,C-1$ 。如图22中所示,用输入端口2215、输出端口2217和LFSR单元2200来构造所述电路,其中,输入端口2215用于接收信息数据,输出端口2217用于输出信息数据和CRC,LFSR单元2200连接在输入端口2215和输出端口2217之间。LFSR单元2200包括L个移位寄存器。所述电路还构造有L个状态寄存器2213,所述状态寄存器2213连接到L个移位寄存器中的相应的移位寄存器,以将数据值写入所述L个移位寄存器以及从所述L个移位寄存器读取数据值。第一开关2201位于输入端口2215和LFSR单元2200之间。第一开关2201具有位置X和位置Y,其中,位置X使输入端口2215与LFSR单元2200连接,位置Y使输入端口2215与LFSR单元2200之间断开连接。第二开关2203位于LFSR单元2200的反馈回路。第二开关2203具有连接LFSR单元2200的反馈回路的位置X和与LFSR单元2200的反馈回路断开连接的位置Y。第三开关2205位于输入端口2215、LFSR单元2200和输出端口2217之间。第三开关2205具有将输入端口2215与输出端口2217连接的位置X以及使LFSR单元2200与输出端口2217连接的位置Y。该电路实现与图21中所示电路相同的CRC计算。相应过程概述如下:

[0144] 将LFSR单元2200初始化为全零状态;设置 $k=0$;将状态寄存器2213初始化为零;将所有开关2201、2203、2205和2207均设置在位置X。

[0145] 经输入端口2215将 $b_k(x)$ 一次一位地输入到电路中。需要注意的是,对于每一位输入,LFSR也移位一次。

[0146] 将LFSR单元2200中的移位寄存器的值写入相应的状态寄存器2213;将所有开关2201、2203、2205和2207均改换到位置Y。

[0147] 使LFSR单元2200移位L次,以获得作为L位CRC的 $p_k(x)$ 。

[0148] 将所有开关2201、2203、2205和2207均设置在位置X;将状态寄存器2213的值写入LFSR单元2200中的相应移位寄存器中。

[0149] $K+1$ 。

[0150] 如果 $k < C$,则转到步骤2。

[0151] 根据本发明原理的第九实施例,为多个码块计算CRC的另一方法使用两个LFSR。如

图23所示,用输入端口2311、输出端口2313、第一LFSR单元2300和第二LFSR单元2301来构造所述电路,其中,输入端口2311用于接收信息数据,输出端口2313用于输出信息数据和循环冗余校验,第一LFSR单元2300连接在输入端口2311和输出端口2313之间并且包括L个移位寄存器,第二LFSR单元2301与第一LFSR单元2300并联地连接在输出端口2311和输出端口2313之间并且包括L个移位寄存器。第一开关2303位于输入端口2311与第一LFSR单元2300和第二LFSR单元2301之间的公共节点2317之间。第一开关具有位置X和位置Y,其中,位置X将输入端口2311与公共节点2317连接,位置Y使输入端口2311与公共节点2317断开连接。第二开关2305位于第一LFSR单元2300的反馈回路。第二开关2305具有连接第一LFSR单元2300的反馈回路的位置X和与第一LFSR单元2300的反馈回路断开连接的位置Y。第三开关2307位于输入端口2311、第一LFSR单元2300和第二LFSR单元2301之间的公共节点2317、以及输出端口2313之间。第三开关2307具有将输出端口2311与输出端口2313连接的位置X、使第一LFSR单元2300与输出端口2313连接的位置Y以及使第二LFSR单元2301与输出端口2313连接的位置Z。第四开关2309位于第二LFSR单元2301的反馈回路。第四开关2309具有连接第二LFSR单元2301的反馈回路的位置X和与第二LFSR单元2301的反馈回路断开连接的位置Y。该方法可以概述如下:

[0152] 将第一LFSR单元2300和第二LFSR单元2301初始化为全零状态;设置 $k=0$;将所有开关2303、2305、2307和2309均设置在位置X。

[0153] 经输入端口2311将 $b_k(x)$ 一次一位地输入到电路中。需要注意的是,对于每一位输入,第一LFSR单元2300和第二LFSR单元2301也均移位一次。

[0154] 将第一开关2301改变到位置Y。

[0155] 如果 $k=C-1$,则转到步骤8;否则,将开关2305和2307改换到位置Y。

[0156] 使第一LFSR单元2300移位L次,以获得作为第k码块的L位CRC的 $p_k(x)$ 。

[0157] $k+1$,转到步骤2。

[0158] 将开关2305和2307改换到位置Z。

[0159] 使第二LFSR单元2301移位L次,以获得作为最后一个码块的L位CRC的 $p_{C-1}(x)$ 。

[0160] 除了最后一个码块之外,该方法仅基于第k码块中的信息位来计算第k码块CRC。因此,除了最后一个码块CRC之外,第k码块CRC可被表示为:

[0161] $p_k(x) = b_k(x) \cdot x^{L-q_k(x)} \cdot g(x)$, 对于 $k=0, 1, \dots, C-2$, (23)

[0162] 其中,C是码块的总数。通过LFSR2来计算最后一个码块CRC,最后一个码块CRC得自所有码块中的信息位。因此,最后一个码块CRC可被表示为:

[0163] $p_k(x) = a_k(x) \cdot x^{L-q_k(x)} \cdot g(x)$, 对于 $k=C-1$ 。 (24)

[0164] 根据本发明原理的第十实施例,另一方法是:在将消息 $a(x)$ 输入到CRC计算电路之前,在所有码块CRC的位位置处将L个0插入到消息 $a(x)$ 中。图24示出了该实现方式的一个示例。用输入端口2407、输出端口2409和LFSR单元2400来构造所述电路,其中,输入端口2407用于接收信息数据,输出端口2409用于输出信息数据和CRC,LFSR单元2400连接在输入端口2407和输出端口2409之间并且包括L个移位寄存器,用于用具有L-1阶的循环冗余校验生成多项式 $g(x)$ 来变换信息数据。第一开关2401位于输入端口2407和LFSR单元2400之间。第一开关2401具有位置X和位置Y,其中,位置X使输入端口2407与LFSR单元2400连接,位置Y使输入端口2407与LFSR单元2400之间断开连接。第二开关2403位于LFSR单元2400的反馈回路。

第二开关2403具有连接LFSR单元2400的反馈回路的位置X和与LFSR单元2400的反馈回路断开连接的位置Y。第三开关2405位于输入端口2407、LFSR单元2400和输出端口2409之间。第三开关2405具有将输入端口2407与输出端口2409连接的位置X以及使LFSR单元2400与输出端口2409连接的位置Y。需要注意的是,通过将开关的位置从X改换到Y,以进行L次移位,来隐含地添加L个0。在这种情况下,本质上,我们允许LFSR的初始状态依赖于先前的码块,因此使当前的CRC能够保护当前码块以及先前码块中的位。该方法可概述如下:

[0165] 将LFSR单元2400初始化为全零状态;设置 $k=0$;将所有开关2401、2403和2405均设置在位置X。

[0166] 经输入端口2407将 $b_k(x)$ 一次一位地输入到电路中。需要注意的是,对于每一位输入,LFSR也移位一次。

[0167] 将所有开关2401、2403和2405均改换到位置Y。

[0168] 使LFSR单元2400移位L次,以获得作为第k码块的L位CRC的 $p_k(x)$ 。

[0169] 将所有开关改换到位置X。

[0170] $k+1$ 。

[0171] 如果 $k < C$,则转到步骤2。

[0172] 尽管已经结合优选实施例显示和描述了本发明,但是对于本领域技术人员明显的是,在不脱离由权利要求限定的本发明的精神和范围的情况下,可进行修改和变化。

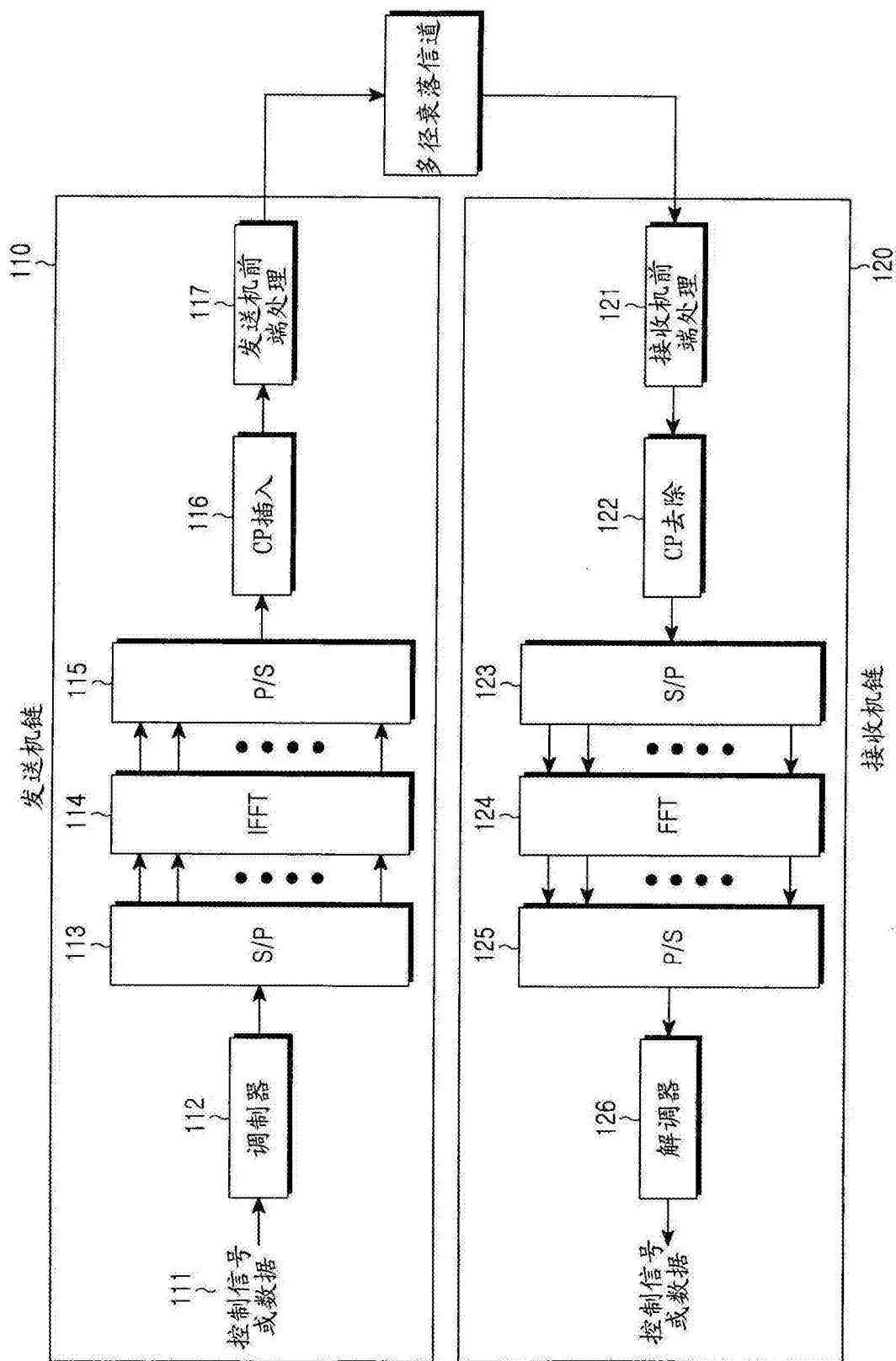


图1

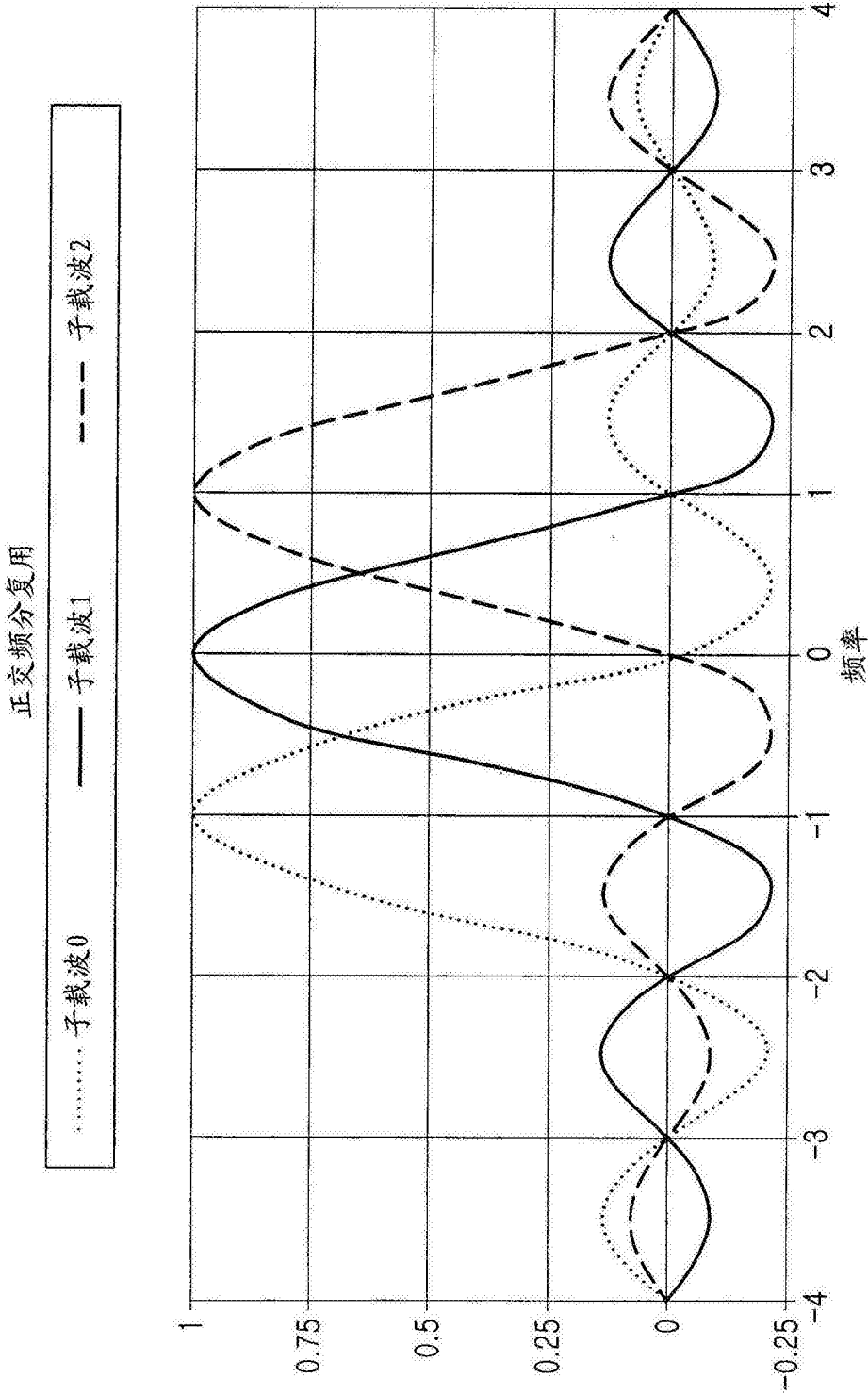


图2

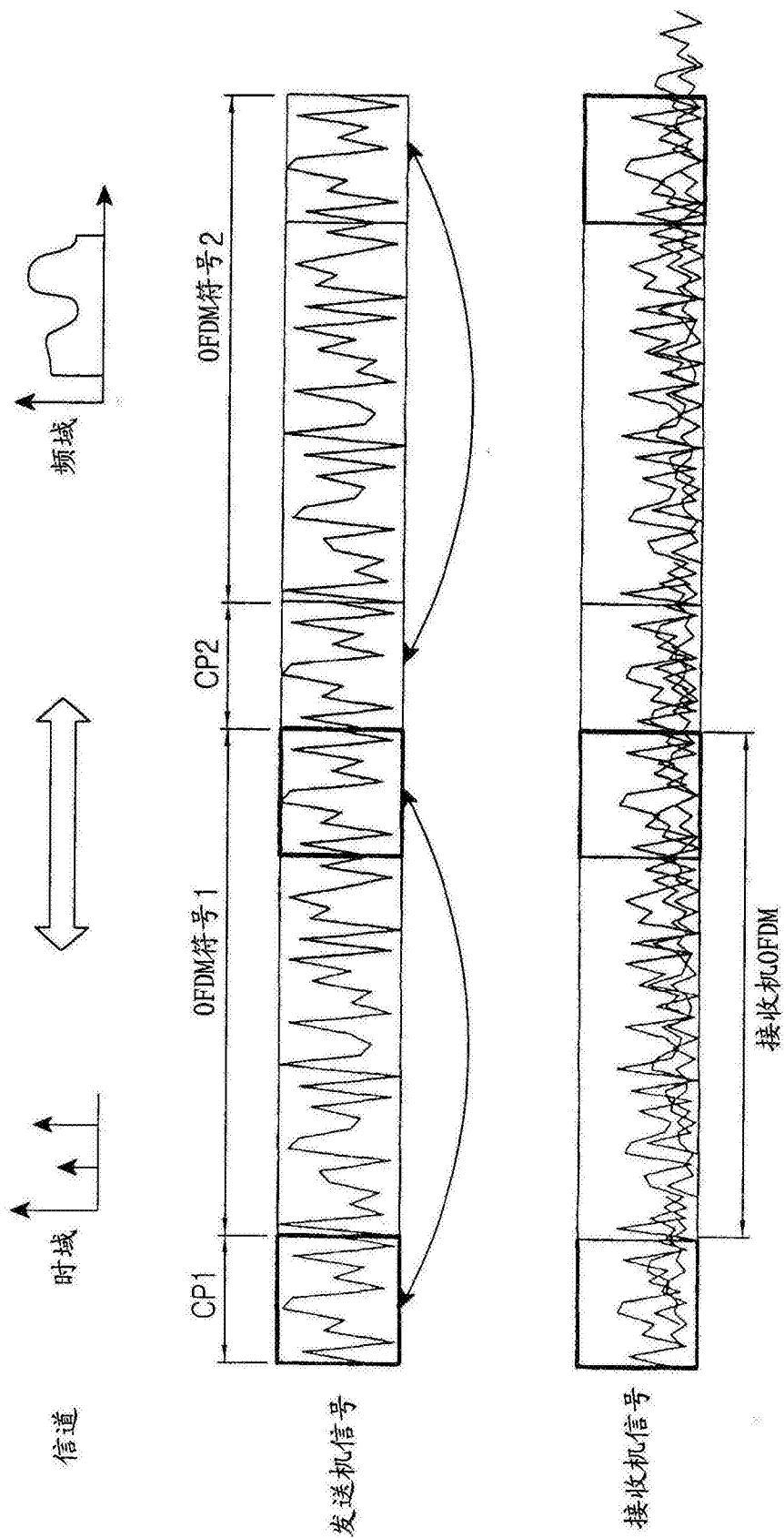


图3

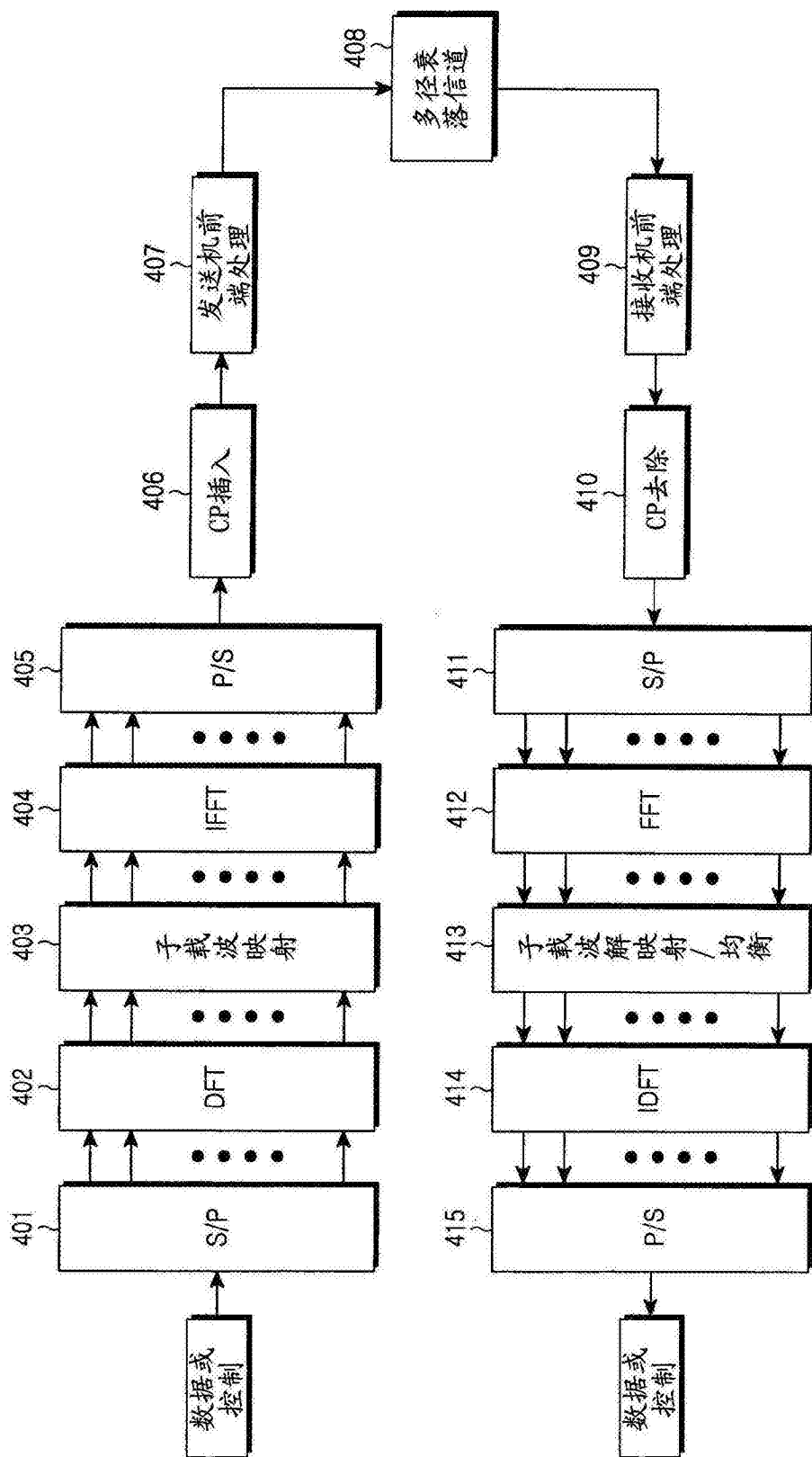


图4

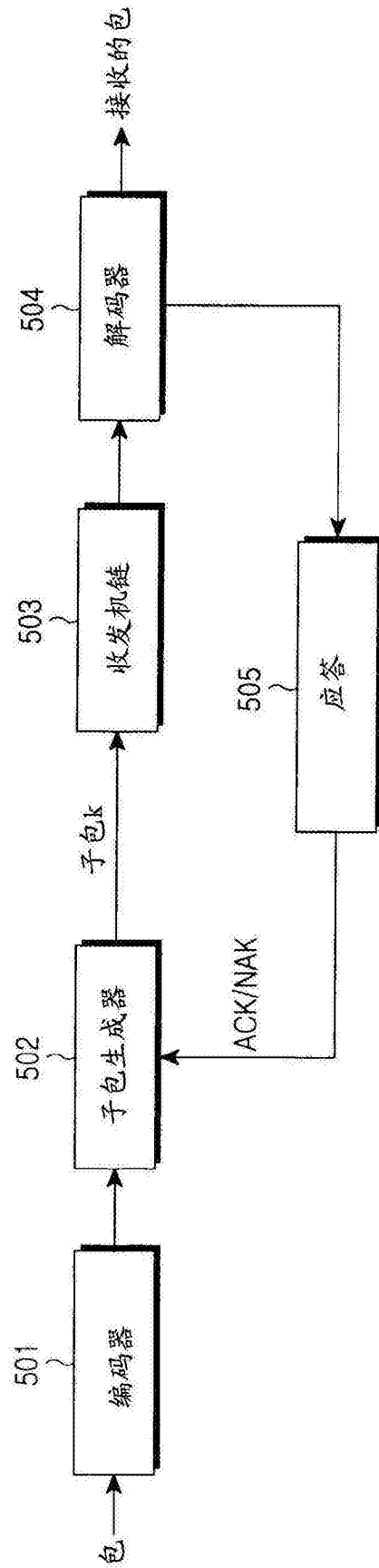


图5

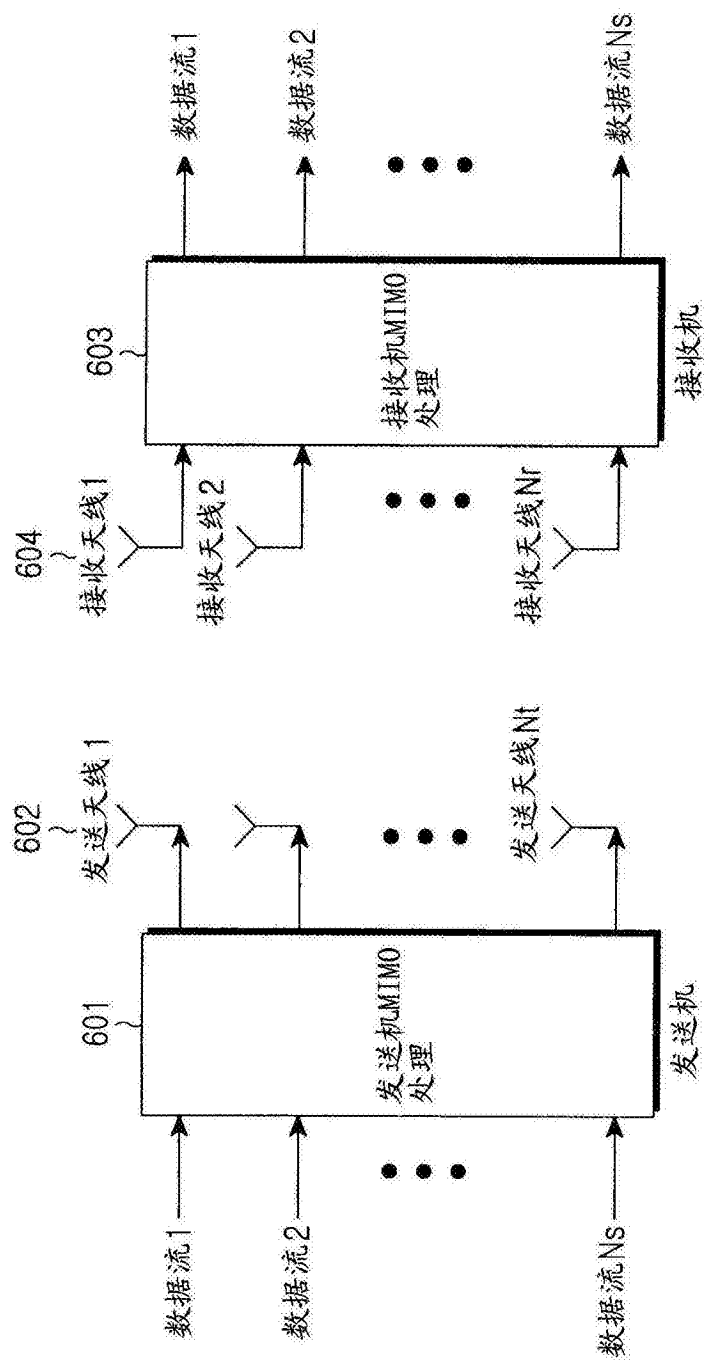


图6

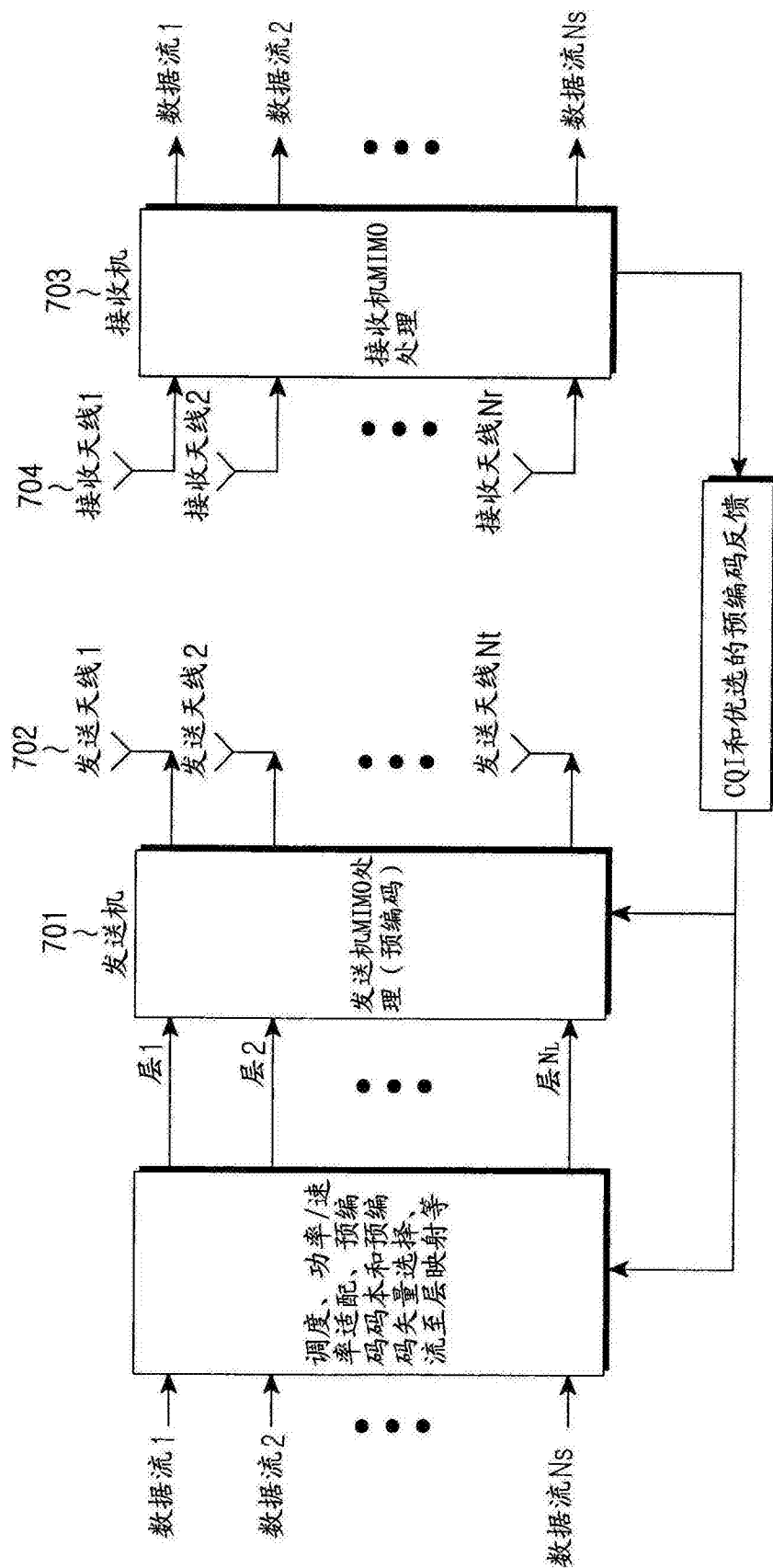


图7

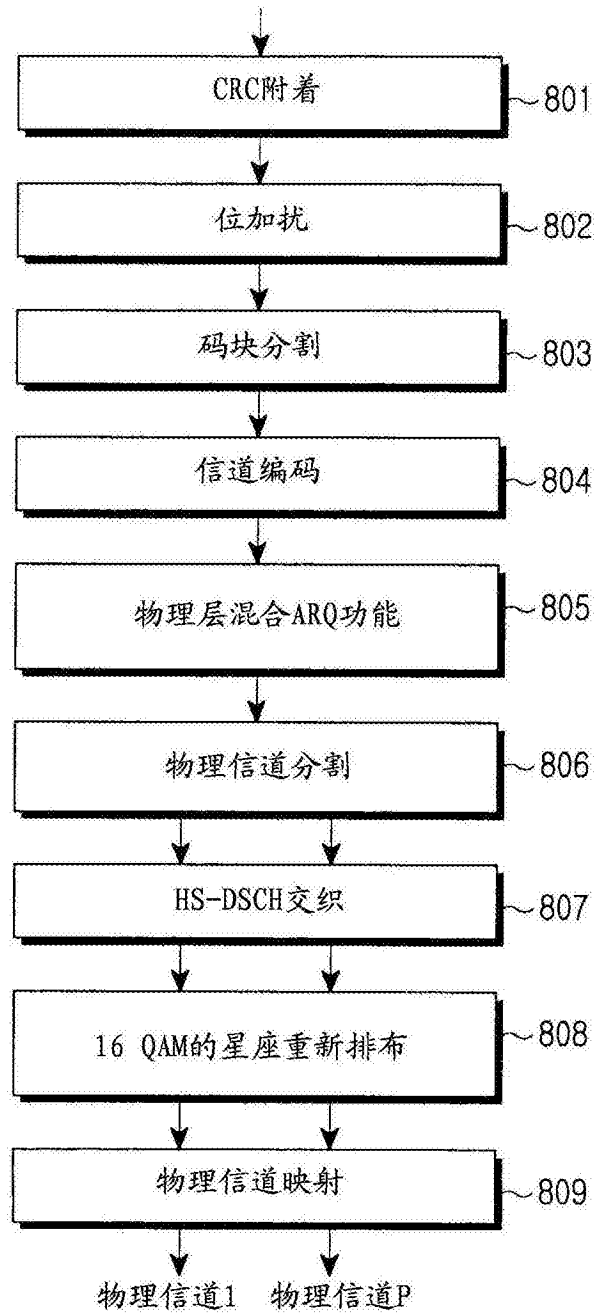


图8

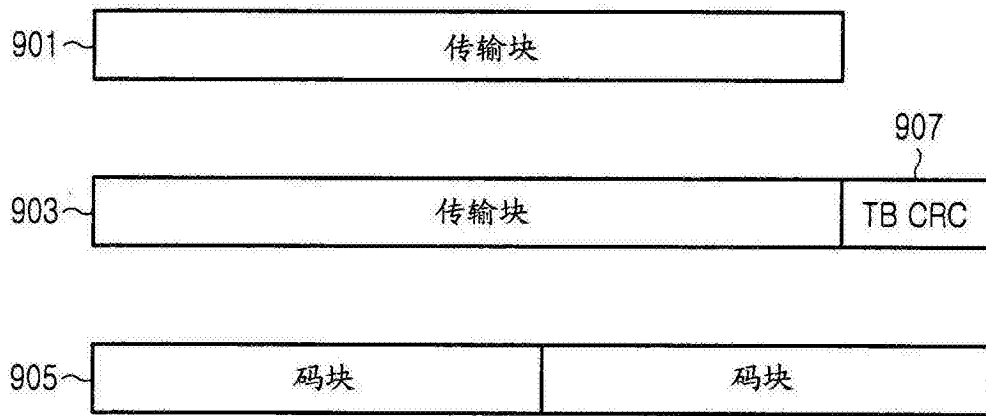


图9

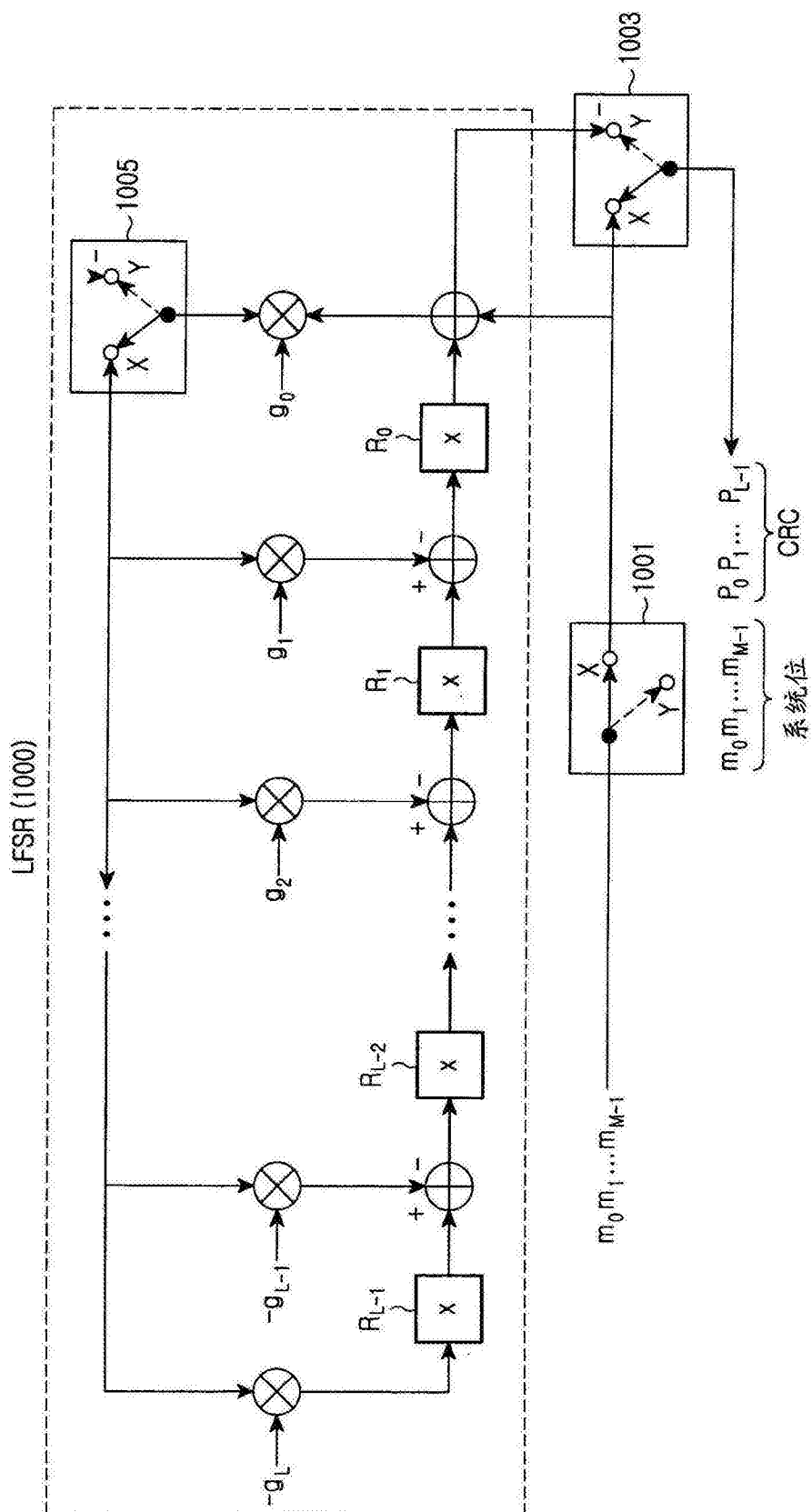


图10

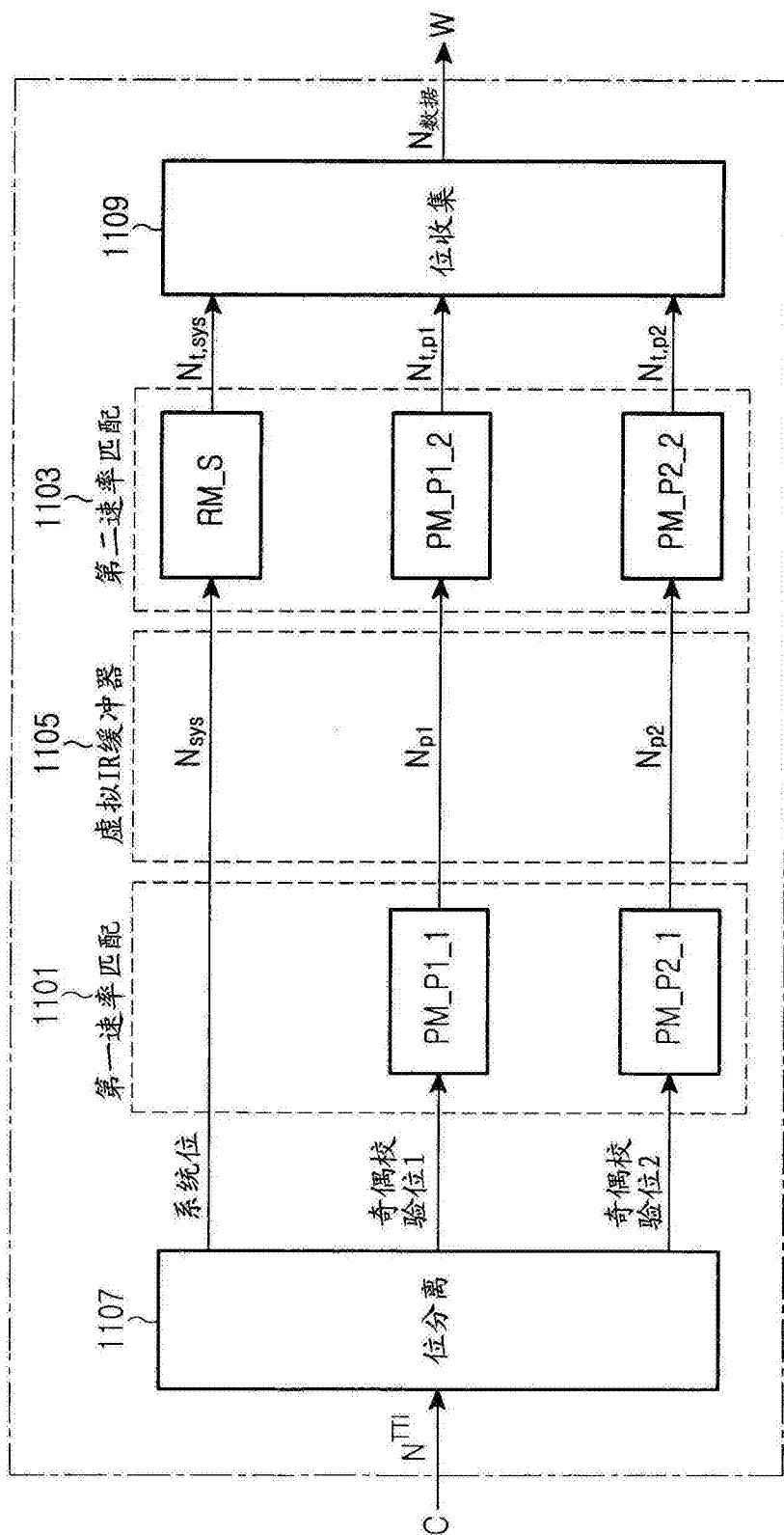


图11

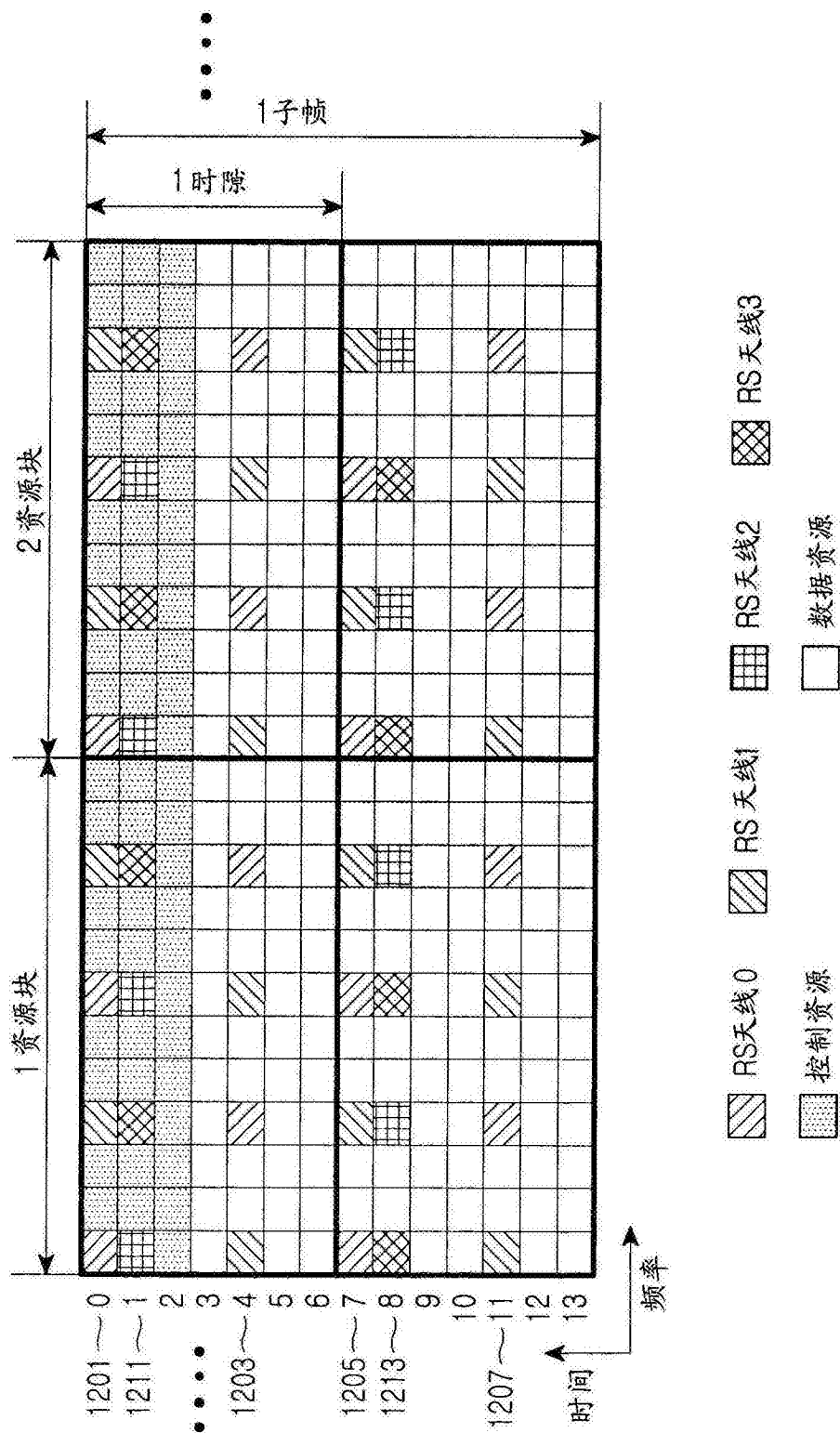


图12

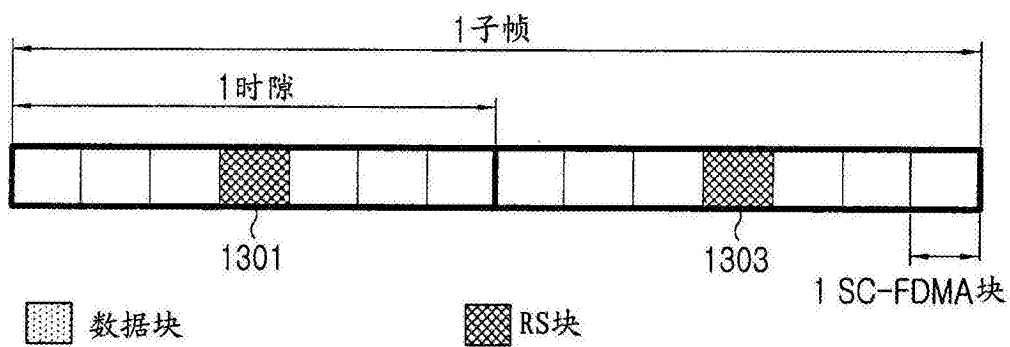


图13

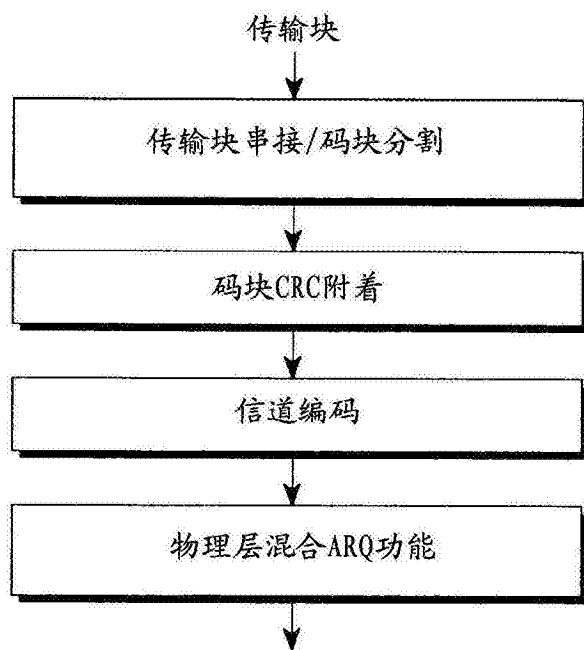


图14

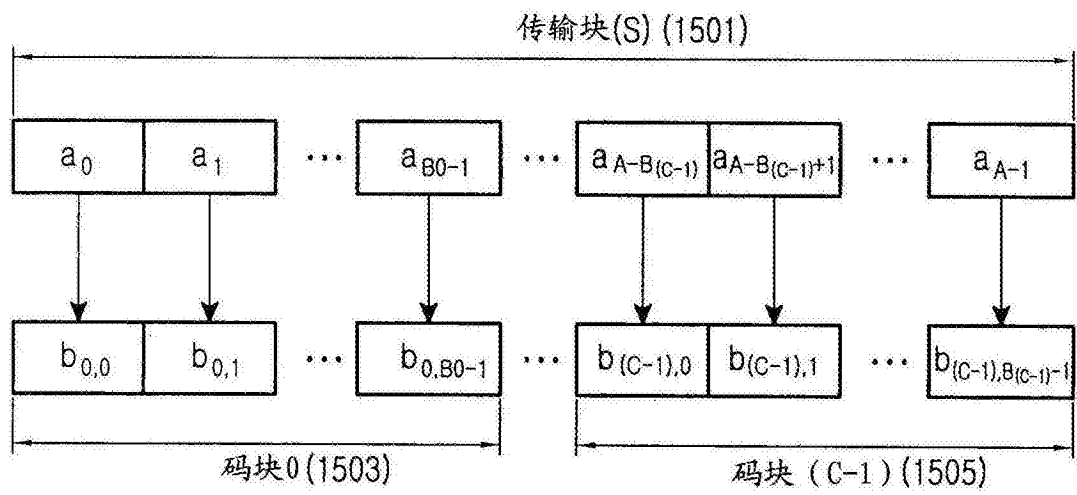


图15

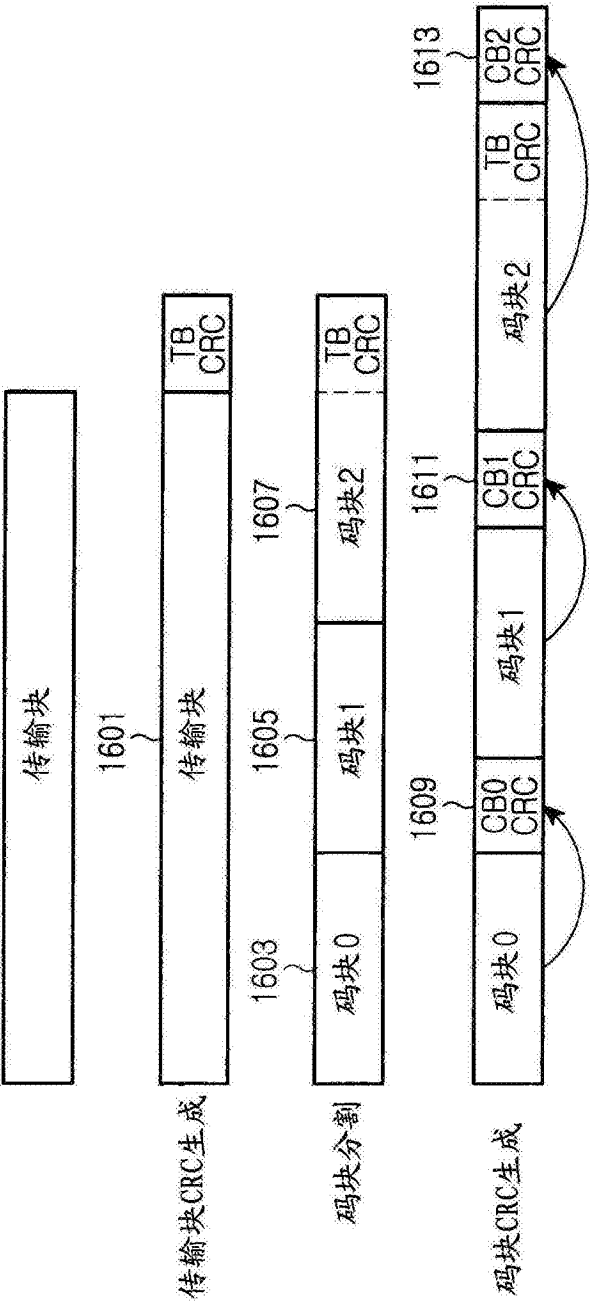


图16

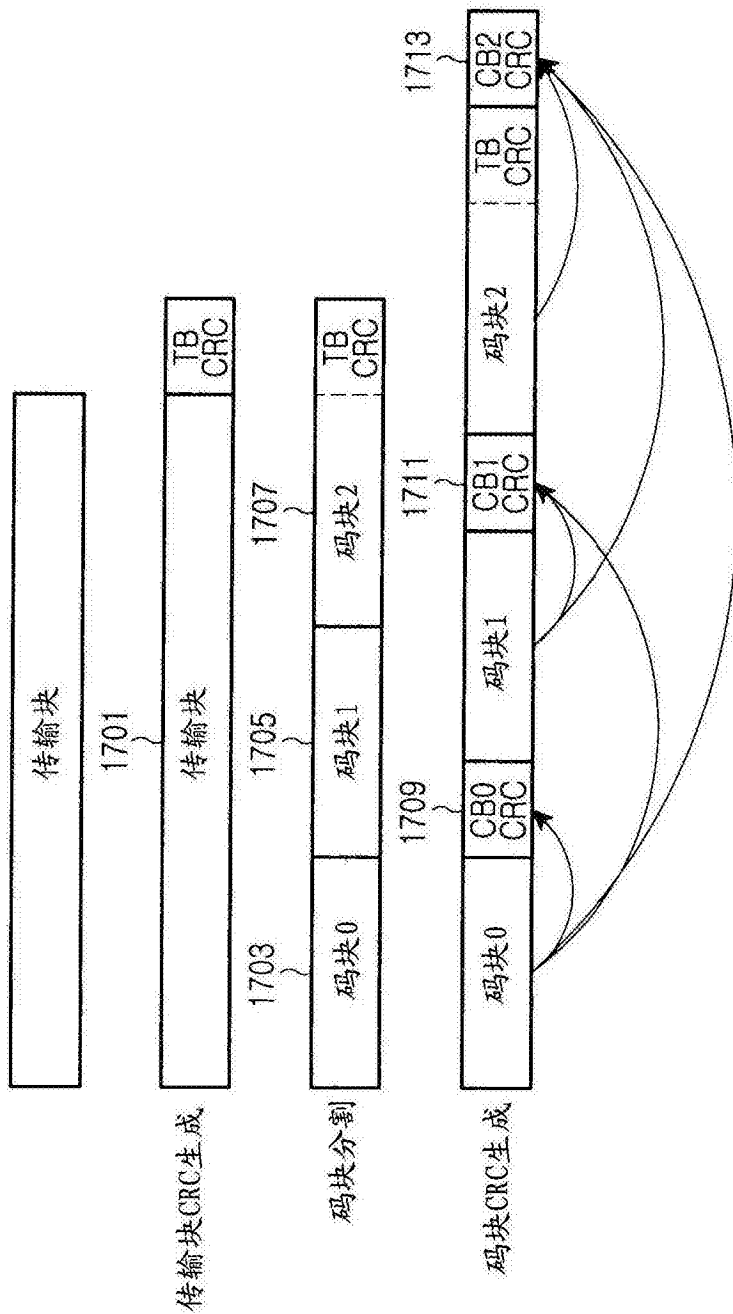


图17

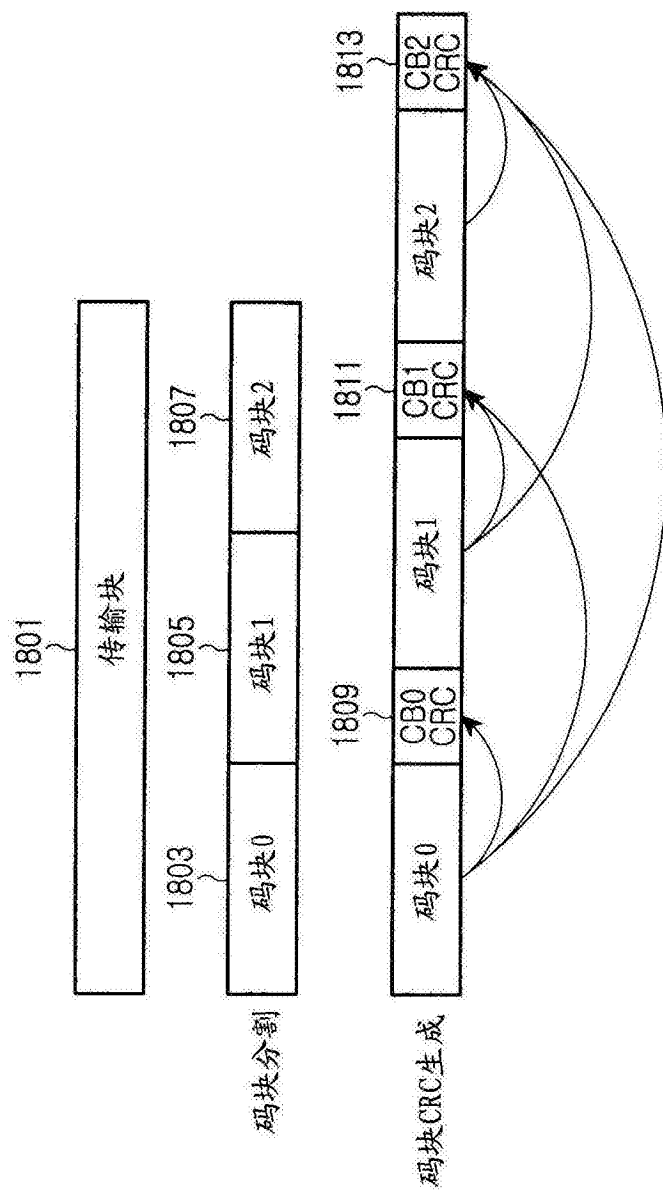


图18

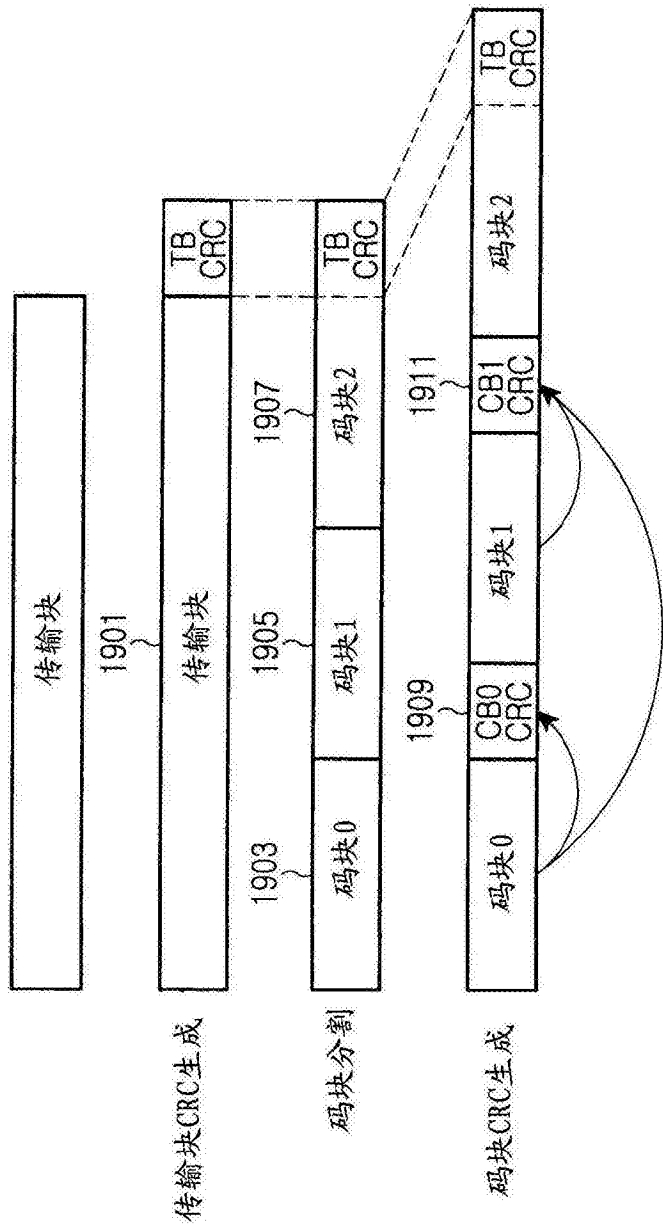


图19

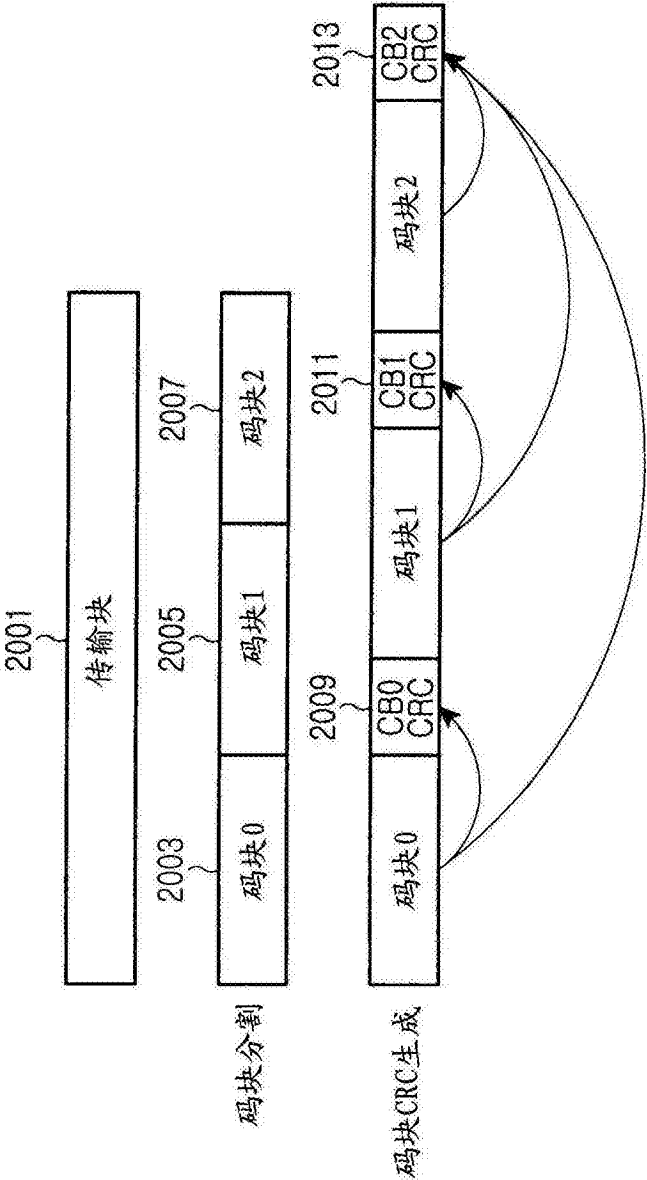


图20

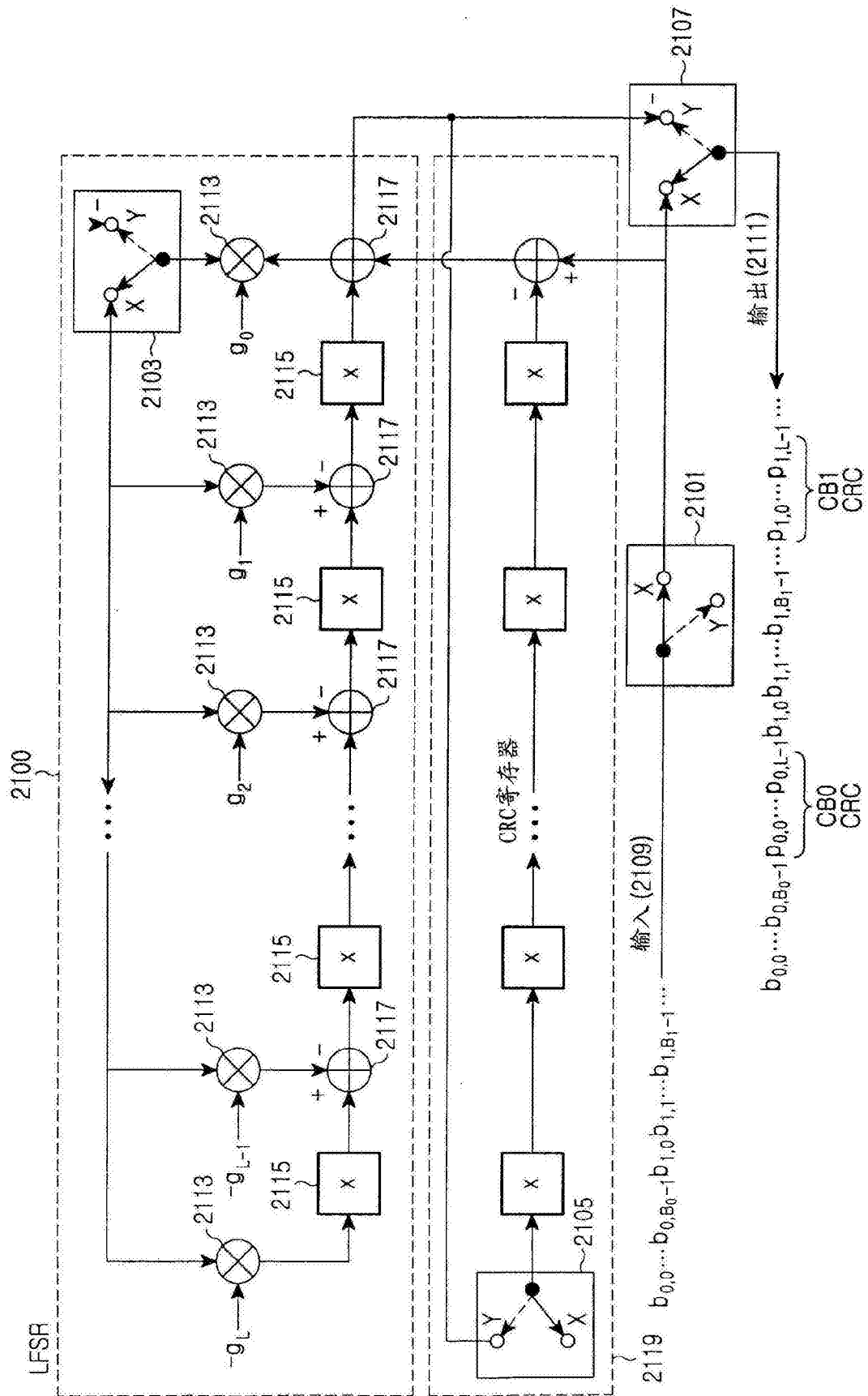


图21

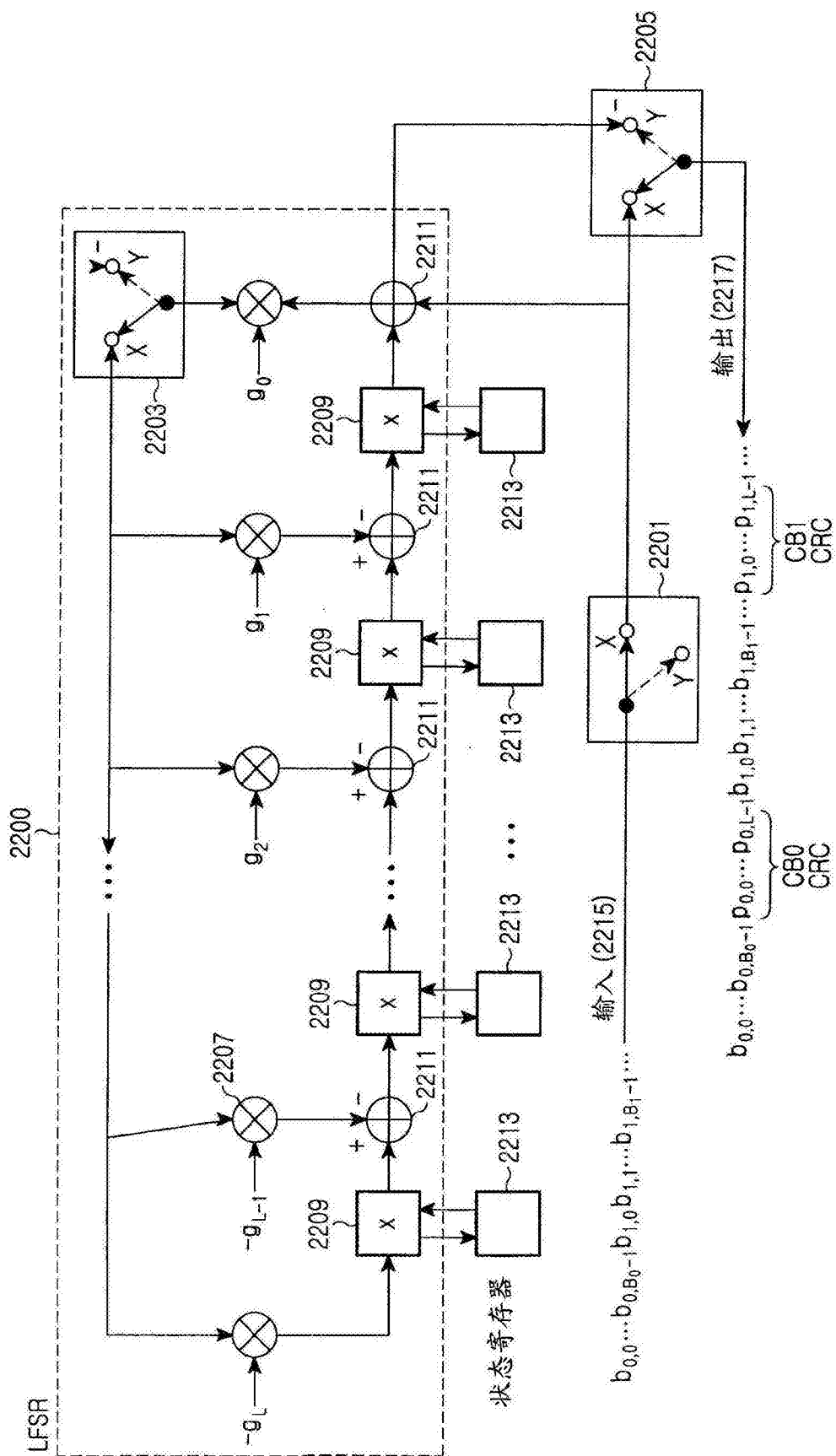


图22

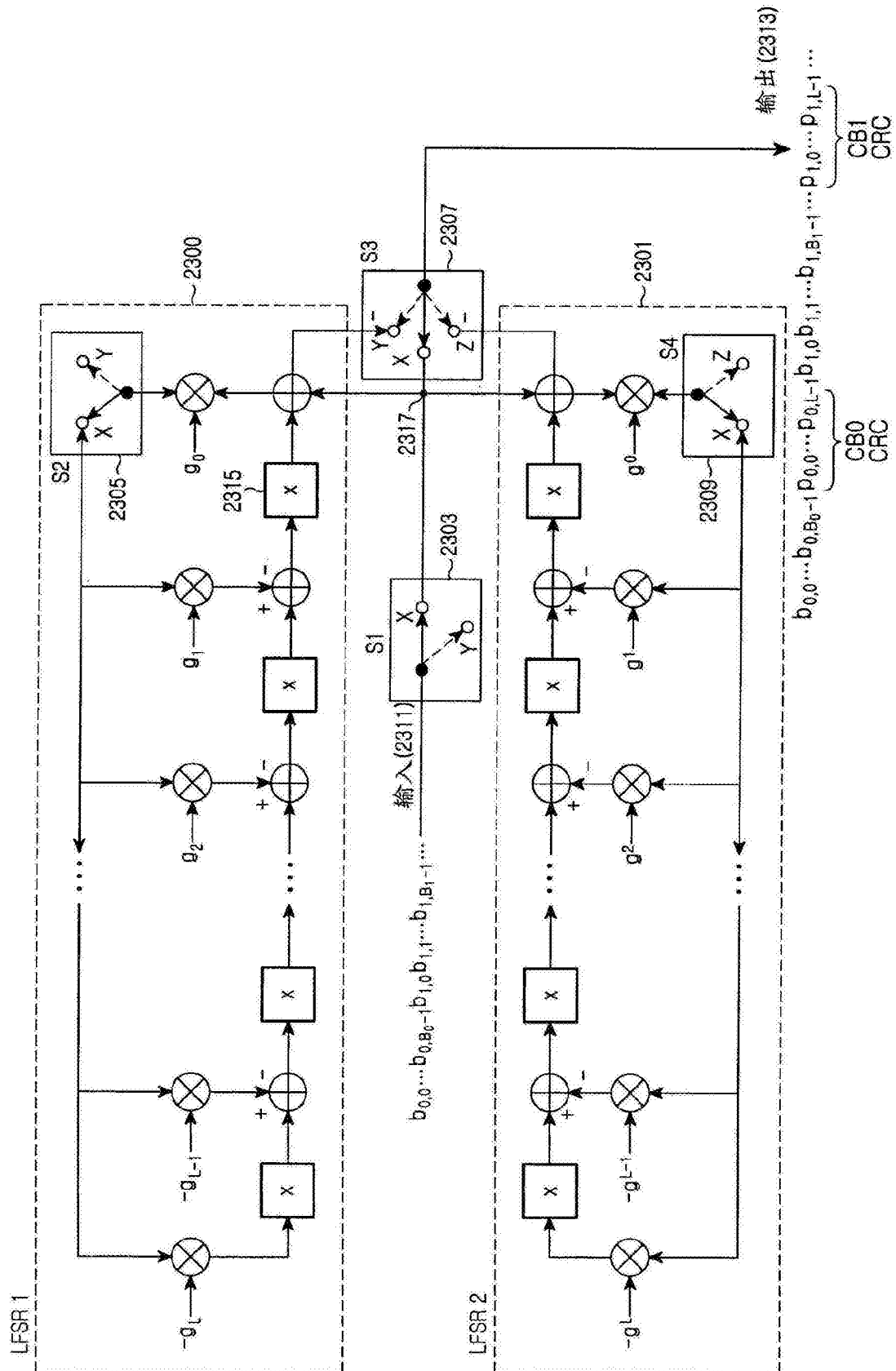


图23

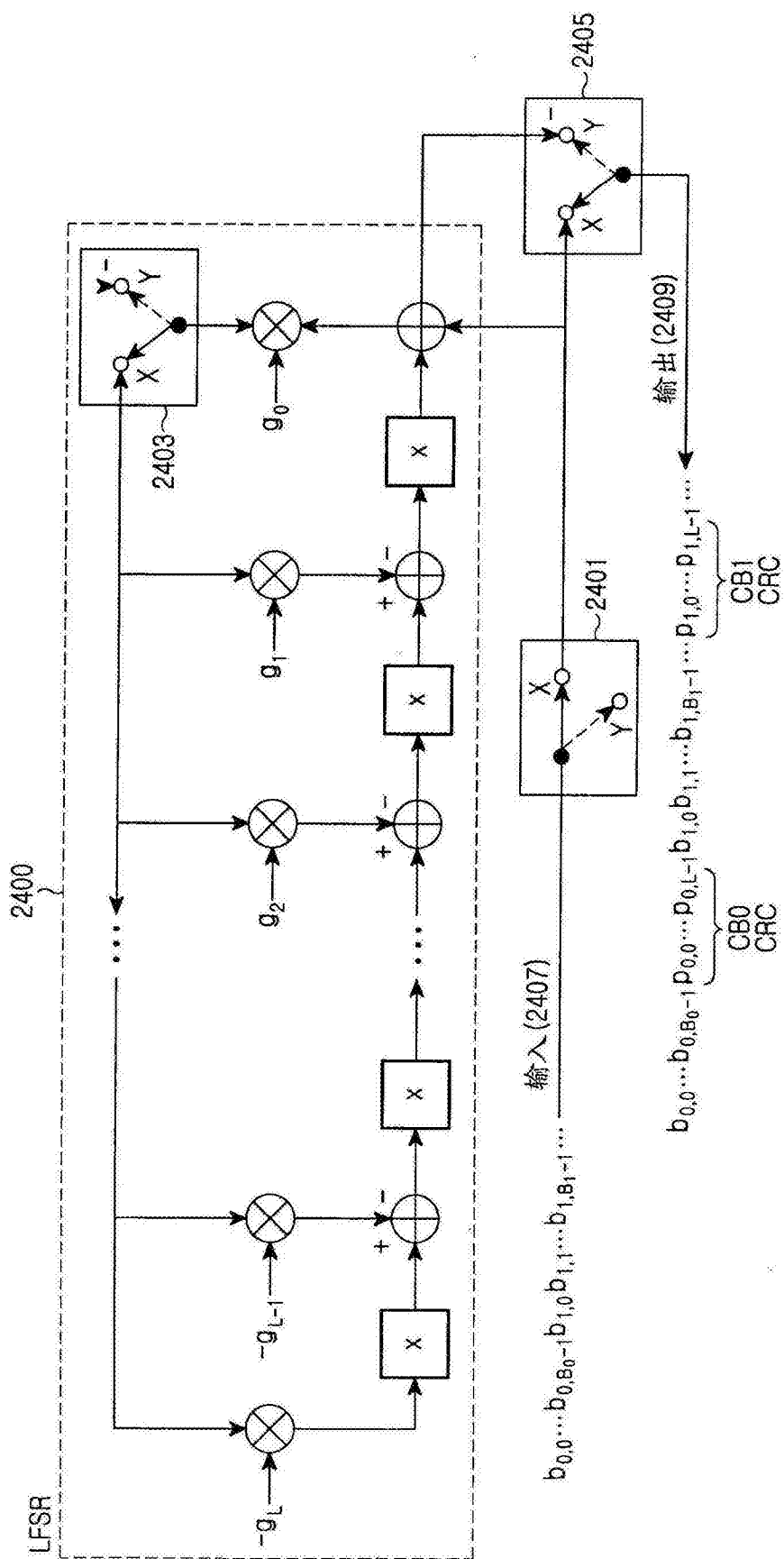


图24