



República Federativa do Brasil  
Ministério da Economia  
Instituto Nacional da Propriedade Industrial

**(11) BR 112015019409-5 B1**



**(22) Data do Depósito:** 13/02/2014

**(45) Data de Concessão:** 11/01/2022

**(54) Título:** PROJETO DE LDPC UTILIZANDO CONSTRUÇÕES QUASE-CÍCLICAS E PERFURAÇÃO PARA ALTA TAXA, ALTO PARALELISMO, E PISO DE ERRO BAIXO

**(51) Int.Cl.:** H03M 13/11; H03M 13/03.

**(30) Prioridade Unionista:** 13/02/2013 US 61/764,476; 13/02/2014 US 14/179,871; 13/02/2014 US 14/179,942.

**(73) Titular(es):** QUALCOMM INCORPORATED.

**(72) Inventor(es):** THOMAS J. RICHARDSON.

**(86) Pedido PCT:** PCT US2014016261 de 13/02/2014

**(87) Publicação PCT:** WO 2014/127129 de 21/08/2014

**(85) Data do Início da Fase Nacional:** 12/08/2015

**(57) Resumo:** PROJETO DE LDPC UTILIZANDO CONSTRUÇÕES QUASE-CÍCLICAS E PERFURAÇÃO PARA ALTA TAXA, ALTO PARALELISMO, E PISO DE ERRO BAIXO. Um método de codificação de dados é divulgado. Um codificador recebe um conjunto de bits de informação e realiza uma operação de codificação de LDPC elevado nos bits de informação para produzir uma palavra-código. O codificador então perfura todos os bits elevados da palavra-código que correspondem a um ou mais bits base perfurados de um código de LDPC da base usado para a operação de codificação de LDPC. O código de LDPC da base não tem múltiplas bordas, e os um ou mais bits de base perfurados são aqueles que correspondem a um ou mais nós base perfurados, respectivamente, do código de LDPC da base. Para algumas modalidades, os um ou mais nós base perfurados correspondem a um ou mais nós variáveis de grau 2. O decodificador LDPC trata os bits de palavras código perfurados como apagados durante a operação e decodificação iterativa.

**PROJETO DE LDPC UTILIZANDO CONSTRUÇÕES QUASE-CÍCLICAS E  
PERFURAÇÃO PARA ALTA TAXA, ALTO PARALELISMO, E PISO DE ERRO  
BAIXO.**

CAMPO TÉCNICO

[0001] As presentes modalidades referem-se geralmente aos sistemas de comunicação e de armazenamento de dados, e, especificamente, aos sistemas de comunicações e de armazenamento de dados que usam códigos de LDPC.

FUNDAMENTOS DA TÉCNICA RELACIONADA

[0002] Muitos sistemas de comunicação utilizam códigos de correção de erros. Especificamente, os códigos de correção de erro para compensar a falta de fiabilidade intrínseca da transferência de informação nestes sistemas através da introdução de redundância no fluxo de dados. Códigos de verificação de paridade de baixa densidade (LDPC) são um tipo especial de códigos de correção de erros que usam um sistema de codificação iterativo. Os códigos de LDPC podem ser representados por gráficos bipartidos (muitas vezes referidos como "gráficos de Tanner"), em que um conjunto de nós variáveis corresponde aos bits de uma palavra-código, e um conjunto de nós de verificação corresponde a um conjunto de restrições de verificação de paridade que definem o código. Um nó variável e um nó de verificação são considerados "vizinhos" se eles estão conectados por uma borda no gráfico. Uma sequência de bits tendo uma associação de um-para-um com a sequência de nó variável é uma palavra-código válida, se e apenas se, para cada nó de verificação, os bits associado a todos os nós vizinhos variáveis somam para zero módulo dois (ou seja, que incluem um mesmo número do 1°).

[0003] Por exemplo, a FIG. 1A mostra um gráfico bipartido 100 representando um código de LDPC exemplar. O gráfico bipartido 100 inclui um conjunto de 5

nós variáveis 110 (representados por círculos) conectados a 4 nós de verificação 120 (representados por quadrados). As bordas do gráfico 100 conectam os nós variáveis 110 aos nós de verificação 120. A FIG. 1B mostra uma representação de matriz 150 do gráfico bipartido 100. A representação de matriz 150 inclui uma matriz de verificação de paridade  $H$  e um vetor  $x$  da palavra-código, onde  $x_1$ - $x_5$  representam bits da palavra-código  $x$ . Mais especificamente, o vetor  $x$  da palavra-código representa uma palavra-código válida, se e somente se  $Hx = 0$ . A FIG. 2 ilustra graficamente o efeito de fazer três cópias do gráfico da FIG. 1A, por exemplo, como descrito na Patente dos EUA de propriedade comum 7.552.097. Três cópias podem ser interligadas pela permutação de bordas iguais entre as cópias. Se as permutações estão restritas às permutações cíclicas, então, o gráfico resultante corresponde a uma LDPC quase-cíclica com o levantamento  $Z = 3$ . O gráfico original a partir da qual três cópias foram feitas é referido aqui como o gráfico de base.

[0004] A palavra-código de LDPC recebida pode ser decodificada para produzir uma versão reconstruída da palavra-código inicial. Na ausência de erros, ou, no caso de erros corrigíveis, a decodificação pode ser utilizada para recuperar a unidade de dados original que foi codificada. Decodificador(es) de LDPC geralmente operam através da troca de mensagens dentro do gráfico bipartido 100, ao longo das bordas, e atualização destas mensagens através da realização de cálculos nos nós com base nas mensagens recebidas. Por exemplo, cada nó variável 110 no gráfico 100 pode ser inicialmente fornecido com um "bit suave" (por exemplo, representando o bit recebido da palavra-código), que indica uma estimativa do valor do bit associado como determinado por observações a parti do canal

de comunicações. Usando esses bits suaves os decodificadores de LDPC podem atualizar as mensagens iterativamente pela leitura delas ou alguma parte das mesmas, a partir da memória e escrita de uma mensagem atualizada, ou alguma parte dela, de volta para, a memória. As operações de atualização são normalmente com base nas restrições de verificação de paridade do código de LDPC correspondente. Em implementações para códigos de LDPC elevados, as mensagens em bordas iguais são muitas vezes são processadas em paralelo.

[0005] Os códigos de LDPC concebidos para aplicações de alta velocidade frequentemente usam construções quase-cíclicas com grandes fatores de elevação e gráficos de base relativamente pequenos para suportar o alto paralelismo nas operações de codificação e decodificação. Os códigos de LDPC com taxas de código mais elevadas (por exemplo, a razão entre o comprimento da mensagem  $K$  para o comprimento da palavra-código  $N$ ) tendem a ter relativamente menos verificações de paridade. Se o número de verificações de paridade de base é menor do que o grau de um nó variável (por exemplo, o número de bordas conectadas a um nó variável), então, no gráfico de base, aquele nó variável é ligado a, pelo menos, uma das verificações de paridade de base por duas ou mais bordas (por exemplo, o nó variável pode ter uma "borda dupla"). Tendo um nó de base variável e um nó de verificação de base ligados por duas ou mais bordas é geralmente indesejável para fins de implementação de hardware paralelo. Por exemplo, tais bordas duplas podem resultar em múltiplas operações de leitura e de gravação simultâneas para as mesmas posições de memória, que por sua vez podem criar problemas de coerência de dados. A segmentação das

atualizações de mensagens paralelas pode ser negativamente afetada pela presença de bordas duplas.

#### SUMÁRIO

[0006] Este Sumário é fornecido para introduzir de uma forma simplificada uma seleção de conceitos que são descritos adicionalmente abaixo na Descrição Detalhada. Este Sumário não se destina a identificar os recursos chave ou recursos essenciais da matéria reivindicada, nem se destina a limitar o escopo da matéria reivindicada.

[0007] Um dispositivo e método de operação são descritos que podem auxiliar na codificação e/ou decodificação de palavras de código de verificação de paridade de baixa densidade (LDPC). Observa-se que a adição de um nó variável perfurado (também conhecido como um nó de variável de estado) no desenho do gráfico de base pode efetivamente aumentar o número de verificações no gráfico por um sem alterar os parâmetros velocidade ( $k$  e  $n$ ) do código. Para algumas modalidades, um codificador pode receber um conjunto de bits de informação e executar uma operação de codificação de LDPC nos bits de informação para produzir uma palavra-código. O dispositivo pode então perfurar um conjunto de bits de palavra-código elevados correspondente a um ou mais nós variável de base baseado em um código de LDPC elevado usado para a operação de codificação de LDPC, em que os bits perfurados correspondem com um ou mais nós de variáveis de base perfurados, respectivamente, do gráfico de LDPC de base. Entende-se que os nós variáveis perfurados na descrição gráfica do código podem ser eliminados a partir da descrição por um processo de combinação de nó de verificação operando na matriz de verificação de paridade elevada. Portanto, pelo menos um dos um ou mais nós base perfurados é entendido por eliminar

as bordas múltiplas entre pares de nó do gráfico de base para o código de LDPC elevado quando a eliminação do nó variável perfurado resulta em várias bordas.

[0008] Para algumas modalidades, os um ou mais nós perfurados podem incluir um nó variável que tem um grau igual a, ou inferior a, um número de nós de verificação do código de LDPC. Por exemplo, pelo menos um dos nós perfurados pode ser um nó variável de grau mais elevado do código de LDPC. Em tal modalidade, o elevado grau do nó é muitas vezes desejável para melhorar o desempenho do código. Por exemplo, a perfuração permite maior grau do nó variável, evitando bordas duplas no gráfico de base. A presença do nó variável perfurado no gráfico aumenta efetivamente o número de nós de verificação que, de outro modo estariam presentes em um gráfico base de um código do mesmo tamanho e taxa. Para outras modalidades, pelo menos um dos nós perfurados pode ser um nó variável de grau dois usado para dividir um nó de verificação que, de outro modo, seria ligado a um nó variável do código de LDPC por duas ou mais bordas. Um nó de grau dois perfurado pode ser eliminado da descrição pela adição de duas verificações de paridade ao qual ele está ligado. O pelo menos um nó variável de grau dois de base perfurado pode, assim, ser utilizado para eliminar bordas duplas no gráfico de LDPC de base. Da mesma forma, um nó perfurado de grau elevado pode ser eliminado a partir de uma representação da matriz de verificação de paridade por um processo de eliminação que soma os nós de restrição para reduzir eficazmente o grau do nó variável para um. Um nó perfurado de grau um pode ser eliminado a partir do gráfico juntamente com o seu nó de verificação vizinho, sem alterar o código. Tal processo de eliminação é susceptível a introduzir bordas duplas ou

múltiplas na representação que é indesejável para a execução paralela da decodificação.

[0009] Pela eliminação ou redução de bordas duplas (ou múltiplas) do gráfico de LDPC de base, as presentes modalidades podem reduzir a complexidade do hardware que executa as operações de decodificação de LDPC em paralelo, aumentando assim a eficiência de processamento dos decodificadores de LDPC que implementam os códigos de LDPC elevados. Isto simplifica ainda mais as operações leitura e/ou gravação realizadas na memória, e assegura que as operações de leitura e gravação não são executadas fora de ordem. Ao permitir maiores graus do nó variável, enquanto evita bordas duplas, as presentes modalidades também podem melhorar o desempenho da correção de erro do sistema de codificação de LDPC.

#### BREVE DESCRIÇÃO DOS DESENHOS

[0010] As presentes modalidades são ilustradas por meio de exemplo e não são destinadas a ser limitadas pelas figuras dos desenhos de acompanhamento, onde:

[0011] FIGS. 1A-1B mostram representações gráficas e de matriz de um código de LDPC exemplar;

[0012] FIG. 2 ilustra graficamente o efeito de fazer três cópias do gráfico da FIG. 1A;

[0013] FIG. 3 mostra um sistema de comunicações de acordo com algumas modalidades;

[0014] FIG. 4 é um diagrama em bloco de um dispositivo de comunicações de acordo com algumas modalidades;

[0015] FIG. 5 é um fluxograma ilustrativo que representa uma operação de codificação de LDPC de acordo com algumas modalidades;

[0016] FIG. 6 é um fluxograma ilustrativo que representa uma operação de decodificação de LDPC de acordo com algumas modalidades;

[0017] FIG. 7 mostra uma matriz de verificação de paridade exemplar associada com um código de LDPC com taxa  $r = 27/30$ ;

[0018] FIG. 8 mostra uma matriz de verificação de paridade exemplar associada com um código de LDPC com taxa  $r = 13/15$ ;

[0019] FIG. 9 mostra uma matriz de verificação de paridade exemplar associada com um código de LDPC com taxa  $r = 21/28$ ; e

[0020] FIG. 10 é um diagrama em bloco de um dispositivo de comunicações de acordo com algumas modalidades.

#### DESCRIÇÃO DETALHADA

[0021] Na seguinte descrição, numerosos detalhes específicos são apresentados como exemplos de componentes, circuitos e processos específicos para proporcionar uma compreensão completa da presente divulgação. O termo "acoplado" como usado aqui, significa ligado diretamente ou ligado através de um ou mais componentes ou circuitos interferentes. Além disso, na descrição seguinte e com a finalidade de explicação, uma nomenclatura específica é apresentada para proporcionar uma compreensão profunda das presentes modalidades. No entanto, será evidente para uma pessoa versada na técnica que estes detalhes específicos podem não ser necessários para a prática das presentes modalidades. Em outros casos, circuitos e dispositivos bem conhecidos são mostrados em forma de diagrama de blocos para evitar obscurecer a presente divulgação. Qualquer um dos sinais fornecidos através de vários barramentos aqui descritos pode ser



multiplexados por tempo com outros sinais e fornecidos sobre um ou mais barramentos comuns. Além disso, a interligação entre os elementos do circuito ou blocos de software pode ser mostrada como barramentos ou como linhas de sinal únicas. Cada um dos barramentos pode alternativamente ser uma linha de sinal única, e cada uma das linhas de sinal únicas podem ser, alternativamente, os barramentos, e uma linha única ou barramento pode representar qualquer um ou mais de uma miríade de mecanismos físicos ou lógicos para a comunicação entre os componentes. As presentes modalidades não devem ser interpretadas como limitadas aos exemplos específicos aqui descritos, mas ao invés por incluir no seu escopo todas as modalidades definidas nas reivindicações anexas.

[0022] FIG. 3 mostra um sistema de comunicações 300 de acordo com algumas modalidades. Um transmissor 310 transmite um sinal sobre um canal 320, e um receptor 330 recebe o sinal do canal 320. O transmissor 310 e receptor 330 podem ser, por exemplo, computadores, comutadores, roteadores, hubs, portas, e/ou dispositivos similares. Em algumas modalidades, o canal 320 é remoto. Em outras modalidades, o canal 320 é um link com fio (ex., um cabo coaxial ou outra conexão física).

[0023] As imperfeições de vários componentes no sistema de comunicações 300 podem se tornar fonte de imparidade de sinal, e assim causar a degradação do sinal. Por exemplo, imperfeições no canal 320 podem introduzir a distorção do canal, que pode incluir a distorção linear, efeitos de multi-trajetória, e/ou Ruído Gaussiano Aditivo Branco (AWGN). Para combater a potencial degradação do sinal, o transmissor 310 e o receptor 330 podem incluir codificadores e decodificadores de LDPC. Especificamente, o transmissor 310 pode realizar a codificação de LDPC nos

dados de saída para produzir uma palavra-código que pode ser subsequentemente decodificada pelo receptor 330 (ex., através de uma operação de decodificação de LDPC) para recuperar os dados originais. Para algumas modalidades, o transmissor 310 pode transmitir as palavras-códigos codificadas por LDPC com um ou mais bits "perfurados", por exemplo, com base em um código de LDPC com um ou mais nós variáveis perfurados.

[0024] "Levantamento" permite que códigos de LDPC sejam implementados usando a codificação paralela e/ou implementações de decodificação enquanto também reduz a complexidade tipicamente associada com códigos de LDPC grandes. Mais especificamente, levantamento é uma técnica para a geração de um código de LDPC relativamente grande a partir de múltiplas cópias de um código de base menor. Por exemplo, um código de LDPC elevado pode ser gerado através da produção de um número ( $Z$ ) de cópias paralelas do gráfico base e depois interligação das cópias paralelas através das permutações de grupos de borda de cada cópia do gráfico base. Uma discussão mais detalhada de códigos de LDPC elevadas pode ser encontrada, por exemplo, no livro intitulado, "Modern Coding Theory", publicada em 17 de março de 2008, por Tom Richardson e Ruediger Urbanke, que é aqui incorporado por referência em sua totalidade.

[0025] Por exemplo, quando se processa uma palavra-código com tamanho de elevação  $Z$ , um decodificador de LDPC pode utilizar  $Z$  elementos de processamento para executar operações de verificação de paridade ou de nó variável em todas as  $Z$  bordas de um gráfico elevado simultaneamente. Especificamente, cada operação de verificação de paridade pode envolver a leitura de um valor correspondente de bit suave a partir da memória, combinação do valor de bit suave com outros valores de bit suave

associados com o nó de verificação e gravação de um bit suave de volta para a memória, que resulta da operação do nó de verificação. Bordas duplas no gráfico base podem desencadear a leitura paralela do mesmo valor de bit suave, em um local de memória, duas vezes durante uma única atualização de verificação de paridade paralela. Circuitos adicionais podem, assim, ser necessários para combinar os valores de bit suave que são escritos de volta para a memória, de modo a incorporar adequadamente as duas atualizações. A eliminação de bordas duplas no gráfico base ajuda a evitar esta complexidade adicional.

[0026] Ao eliminar ou reduzir as bordas duplas (e/ou múltiplas) do código de LDPC base, a perfuração pode reduzir a complexidade do hardware que executa operações de nó de verificação paralelas ou de nó variável, aumentando assim a eficiência do processamento paralelo de um decodificador de LDPC correspondente. Isto simplifica ainda mais as operações de leitura e/ou gravação realizadas na memória, e assegura que as operações de leitura e gravação não são executadas fora de ordem.

[0027] A perfuração é o ato de remover bits de uma palavra-código para produzir uma palavra-código mais curta. Assim, nós variáveis perfurados correspondem à bits de palavra-código que não são efetivamente transmitidos. A perfuração de um nó variável em um código de LDPC cria um código encurtado (por exemplo, devido à remoção de um bit), enquanto também remove eficazmente um nó de verificação. Especificamente, para uma representação de matriz de um código de LDPC, incluindo os bits a serem perfurados, onde o nó variável a ser perfurado tem um grau de um (tal representação pode ser possível através combinação de fileira fornecida no que código é apropriado), a perfuração do nó variável remove o bit associado a partir do código e

remove eficazmente o seu nó de verificação vizinho único a partir do gráfico. Como resultado, o número de nós de verificação no gráfico é reduzido por um. Se o comprimento de bloco base transmitido é  $n-p$ , em que  $p$  é o número de colunas perfuradas, e o número de verificações de paridade base é  $m$ , então a taxa é  $(n-m)/(n-p)$ . O tamanho do bloco de informação binário é  $(n-m)*Z$ , e o tamanho do bloco transmitido é  $(n-p)*Z$ . Observe que se aumentarmos  $n$  e  $p$  por 1 nós podemos aumentar  $m$  por 1 e deixar a taxa e o tamanho do bloco inalterados.

[0028] Como exemplo, considere um código de taxa de 0,9 em que o comprimento do bloco base é 30. Sem a perfuração, o número de nós de verificação que seria utilizado para definir o código base é 3, o que resulta em um código (27, 30) (por exemplo,  $K = 27$  bits de mensagem,  $n = 30$  bits de palavras de código). Tal código é propenso a ter bordas duplas (ou mais) ligando pelo menos um nó de verificação a um nó variável (por exemplo, a menos que todos os nós variáveis tenham um grau máximo de 3). No entanto, pode ser desejável ter nós variáveis de maior grau (por exemplo, grau  $\geq 4$ ), por exemplo, para assegurar um piso de erro profundo. Se um nó variável perfurado é introduzido no código de LDPC, aumentando assim o número total de nós variáveis para 31, em seguida, o número de nós de verificação de base aumenta para 4. É agora possível ter nós variáveis de base de grau 4 sem bordas duplas no gráfico base. Notamos, no entanto, que tal código de LDPC ainda é um código (27, 30).

[0029] Na representação do gráfico bipartido de um código de LDPC, um nó variável perfurado de grau dois combina eficazmente os seus dois nós de verificação vizinhos em um nó de verificação único. O nó variável perfurado de grau dois indica efetivamente que os seus dois

nós de verificação vizinhos têm, ausente o nó de grau dois, a mesma paridade. Por conseguinte, nós variáveis perfurados de grau dois podem ser utilizados para "dividir" os nós de verificação, parecendo assim para aumentar o número total de nós de verificação. Este mecanismo pode, portanto, ser usado para remover bordas múltiplas de um código de LDPC. Um nó variável é tipicamente ligado a, pelo menos, um nó de verificação por duas ou mais bordas, se o grau do nó variável é maior que o número total de nós de verificação ( $N$ ) no gráfico base. Assim, várias bordas em um gráfico base podem ser evitadas e/ou eliminadas através da introdução de um ou mais nós variáveis de grau dois (isto é, assumindo que, pelo menos, um nó variável no gráfico base tem um grau maior do que  $N$ ).

[0030] A perfuração de um nó variável base de elevado grau do código de LDPC também pode aumentar o número de nós de verificação. Além disso, nós de verificação de alto grau podem ser desejáveis no design de LDPC de alto desempenho. Por exemplo, o nó de variável de grau mais elevado pode corresponder a um nó variável que tem um grau igual a (ou menor que) o número total de nós de verificação no gráfico base. Tal nó variável de alto grau pode evidentemente estar presente em um gráfico base sem quaisquer bordas duplas. Como descrito em maior detalhe abaixo, um nó variável perfurado é tratado como "apagado" na decodificação. Assim, para os códigos que visam baixas taxas de erro, pode ser desejável evitar que tais nós participem nas estruturas combinatórias (por exemplo, prendendo conjuntos ou palavras-código próximas) que dão origem aos eventos de piso de erro. Tendo um alto grau geralmente torna menos provável que um nó irá contribuir para um evento de piso de erro.

[0031] Além disso, nós variáveis perfurados de alto grau no gráfico podem melhorar o desempenho do código. Sabe-se que os nós perfurados em um gráfico podem melhorar o chamado limiar iterativo da estrutura de código. Em modelos de LDPC irregulares padrão (isto é, sem nós variáveis perfurados), limiares podem ser melhorados pelo aumento do grau médio no gráfico bipartido, e, assim, aumentar os graus dos nós variáveis e de verificação. Com nós variáveis perfurados, o mesmo efeito pode ser conseguido com menor grau médio, reduzindo assim a complexidade do código de LDPC. Além disso, estruturas de código de LDPC com graus médios mais baixos podem ter um melhor desempenho em gráficos menores. Assim, nós variáveis de alto grau de perfuração podem tanto aumentar o número de nós de verificação (permitindo assim graus mais elevados) quanto melhorar o desempenho dos códigos com graus de nó variável máximos limitados.

[0032] FIG. 4 é um diagrama em bloco de um dispositivo de comunicações 400 de acordo com algumas modalidades. O dispositivo de comunicações 400 inclui um codificador 410, um decodificador 420, e um transceptor 430, que transmite e/ou recebe palavras-códigos codificadas por LDPC através de um canal de comunicações. O codificador 410 inclui uma memória 412, e um codificador de LDPC 414. A memória 412 pode ser usada para armazenar dados (i.e., bits de informação) a serem codificados pelo codificador de LDPC 414. O codificador de LDPC 414 processa os bits de informação armazenados na memória 412 pela geração de palavras-código, com base em um código de LDPC, a ser transmitido para outro dispositivo.

[0033] Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado. Adicionalmente, para algumas modalidades, o código de LDPC da base pode

incluir um ou mais nós perfurados. O codificador de LDPC 414 pode então perfurar um ou mais bits da palavra-código que corresponde aos respectivos nós perfurados do código de LDPC da base. Estes bits de palavra-código perfurados não são transmitidos pelo transceptor 430. Para algumas modalidades, os nós perfurados podem incluir um nó variável de base tendo um grau igual a, ou menor que, um número de nós de verificação do código de LDPC. Por exemplo, pelo menos um dos nós perfurados pode ser um nó variável de grau mais alto do código de LDPC. Para outras modalidades, pelo menos um dos nós perfurados pode ser usado para dividir um nó de verificação que é conectado a um nó variável do código de LDPC por duas ou mais bordas. Tal nó perfurado pode ser usado para eliminar bordas duplas no gráfico base para o código de LDPC elevado.

[0034] O decodificador 420 inclui uma memória 422 e um decodificador de LDPC 424. A memória 422 armazena as palavras-código, recebidas através do transceptor 430, para serem decodificadas pelo decodificador de LDPC 424. O decodificador de LDPC 424 processa as palavras-código armazenados na memória 424 pela realização iterativamente de operações de verificação de paridade, usando um código de LDPC, e tentativa de corrigir quaisquer bits que podem ter sido recebidos no erro. Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado. Ainda, para algumas modalidades, a palavra-código recebida pode incluir um ou mais bits de perfuração como determinado, por exemplo, com base em um conjunto de nós perfurados do código de LDPC correspondente. Como descrito acima, com relação à FIG. 3, os nós perfurados podem ser determinados com base nos graus dos nós variáveis do código de LDPC. O decodificador de LDPC 424 pode então tratar estes nós perfurados como apagados para fins de decodificação. Por

exemplo, o decodificador de LDPC 424 pode definir os logaritmos da razão de verossimilhança (LLRs) dos nós perfurados para zero na inicialização.

[0035] Para algumas modalidades, o decodificador de LDPC 424 pode incluir uma pluralidade de elementos de processamento para realizar as operações de verificação de paridade ou de nó variável em paralelo. Por exemplo, ao processar uma palavra-código com tamanho de elevação  $Z$ , o decodificador de LDPC 424 pode usar um número ( $Z$ ) de elementos de processamento para realizar operações de verificação de paridade em todas as  $Z$  bordas de um gráfico elevado, simultaneamente. Especificamente, cada operação de verificação de paridade pode envolver a leitura de um valor de bit suave correspondente da memória 422, combinando o valor de bit suave com outros valores de bit suave associados com o nó de verificação e gravação de um bit suave de volta para a memória 422 que resulta da operação do nó de verificação. Uma borda dupla em um código de LDPC da base pode desencadear a leitura paralela do mesmo local de memória do valor do bit suave duas vezes durante uma atualização de verificação de paridade única. Assim, circuitos adicionais são tipicamente necessários para combinar os valores de bit suave que são gravados de volta para a memória, de modo a incorporar adequadamente ambas as atualizações. No entanto, a eliminação das bordas duplas no código de LDPC, por exemplo, como descrito acima com relação às FIG. 3, ajuda a evitar esta complexidade extra.

[0036] FIG. 5 é um fluxograma ilustrativo que representa uma operação de codificação de LDPC 500 de acordo com algumas modalidades. Com referência, por exemplo, à FIG. 4, o codificador 410 primeiro recebe um conjunto de bits de informação a ser codificado (510). Os



bits de informação podem corresponder aos dados destinados a ser transmitidos para outro dispositivo (ex., um dispositivo de recepção) sobre um canal de comunicações ou rede. Por exemplo, os bits de informação podem ser recebidos de uma unidade de processamento central (CPU) e armazenados na memória 412.

[0037] O codificador 410 pode então realizar uma operação de codificação de LDPC nos bits de informação para produzir uma palavra-código de LDPC (520). Para algumas modalidades, o codificador de LDPC 414 pode codificar os bits de informação nas palavras-código de LDPC com base em um código de LDPC que é compartilhado pelo codificador 410 e um decodificador correspondente (ex., do dispositivo de recepção). Cada palavra-código pode incluir os bits de informação originais, ou uma porção dos mesmos, assim como um conjunto de bits de paridade que podem ser usados (ex., pelo decodificador) para realizar operações de verificação de paridade em e/ou recuperar os bits de informação originais. O codificador 410 pode ainda perfurar um ou mais bits da palavra-código de LDPC com base em nós variáveis base perfurados do código de LDPC (530). Por exemplo, os um ou mais bits de palavra-código perfurados podem corresponder a um ou mais nós variáveis base perfurados, respectivamente, do código de LDPC da base. Especificamente, pelo menos alguns dos nós perfurados são fornecidos para eliminar várias bordas entre pares de nó no gráfico base do código de LDPC elevado. Para algumas modalidades, os nós perfurados podem incluir um nó variável tendo um grau igual a, ou menor que, um número de nós de verificação do código de LDPC. Para outras modalidades, pelo menos um dos nós perfurados pode ser um nó variável de grau 2. Por exemplo, o nó variável de grau 2 pode ser usado para dividir um nó de verificação que estaria de outro modo

conectado a outro nó variável do código de LDPC por duas ou mais bordas. Para algumas modalidades ambos podem ocorrer, especificamente, tanto um nó variável perfurado de grau alto quanto um nó perfurado de grau dois pode ocorrer no gráfico base. Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado. Ainda adicionalmente, o código de LDPC pode ser com base em uma elevação quase cíclica, sendo que as permutações de grupos de borda são permutações cíclicas.

[0038] FIG. 6 é um fluxograma ilustrativo que representa uma operação de decodificação de LDPC 600 de acordo com algumas modalidades. Com referência, por exemplo, à FIG. 4, o decodificador 420 primeiro recebe uma palavra-código de LDPC a ser decodificada (610). A palavra-código de LDPC pode ser recebida de um dispositivo de transmissão, por exemplo, na forma de um sinal de dados modulado por amplitude de quadratura (QAM). Consequentemente, a palavra-código de LDPC pode corresponder a um subconjunto de bits de rotulagem do sinal de dados QAM de-mapeados.

[0039] O decodificador 420 pode identificar um ou mais bits perfurados da palavra-código de LDPC com base em nós base perfurados do código de LDPC (620). Por exemplo, os um ou mais bits de palavra-código perfurados podem corresponder a um ou mais nós base perfurados, respectivamente, do código de LDPC. Como descrito acima, pelo menos alguns dos nós base perfurados são fornecidos para eliminar várias bordas entre pares de nó no gráfico base do código de LDPC elevado. Para algumas modalidades, os nós perfurados podem incluir um nó variável tendo um grau igual a, ou menor que, um número de nós de verificação do código de LDPC. Para outras modalidades, pelo menos um dos nós perfurados pode ser um nó variável de grau 2. Como

descrito acima, o nó variável de grau 2 pode ser usado para dividir um nó de verificação que estaria de outro modo conectado a outro nó variável do código de LDPC por duas ou mais bordas. Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado (ex., com base em um levantamento quase cíclico).

[0040] O decodificador 420 pode então realizar uma operação de decodificação de LDPC na palavra-código recebida para recuperar os bits de informação original (630). Por exemplo, o decodificador de LDPC 424 pode processar a palavra-código pela realização iterativamente de operações de verificação de paridade, usando o código de LDPC, e tentando corrigir quaisquer bits que podem ter sido recebidos no erro. Para algumas modalidades, o decodificador de LDPC 424 pode tratar os bits de palavra-código perfurados como apagados durante a operação de decodificação, por exemplo, pela definição de LLRs dos nós perfurados para zero na inicialização.

[0041] Nas presentes modalidades, cada um dos códigos de LDPC pode ser visto como uma matriz binária bidimensional de tamanho  $Z \times n$ , onde  $n$  é o comprimento do bloco base (transmissão). Para algumas modalidades, os códigos a jusante propostos são definidos de modo que  $Z=360$ . EM cada constelação,  $k$  bits podem ser tomados de cada vez, por dimensão (ex., para 1024QAM,  $k=5$ ). Além disso,  $k$  é um fator de 360, e  $k$  bits podem ser tomados de cada vez coluna a coluna, gerando assim  $360/k$  dimensões ou  $180/k$  símbolos por coluna. Deve ser então observado que  $k$  é um fator de 60 para o conjunto  $k \in \{1, 2, 3, 4, 5, 6\}$ , nos casos de interesse.

[0042] As FIGs. 7, 8, e 9 mostram matrizes de verificação de paridade exemplares 700, 800, e 900, respectivamente, de acordo com algumas modalidades. Em cada

uma das matrizes de verificação de paridade 700, 800, e 900, a fileira superior fornece os índices para as colunas de  $H$ . A segunda fileira indica as colunas de informação (1) e paridade (0). A terceira fileira indica as colunas transmitidas (1) e colunas perfuradas (0).

[0043] Observe que as matrizes de verificação de paridade 700 e 800, que são associadas com códigos de LDPC com taxas  $r = 27/30$  e  $r = 13/15$ , respectivamente, são sistemáticas. No entanto, a matriz de verificação de paridade 900, que é associada com um código de LDPC com taxa  $r = 21/28$ , tem uma coluna de informação perfurada, e é portanto não completamente sistemática.

[0044] Adicionalmente, a matriz de verificação de paridade 700 tem um nó variável de grau dois perfurado (índice 0). Tal nó pode dividir uma verificação de paridade única em duas. Isto garante que a matriz base não tem bordas duplas, e facilita algumas das modalidades descritas aqui. Uma representação de código equivalente pode ser construída fundindo as duas verificações de paridade e eliminando o nó variável de grau dois perfurado. Além disso, tal representação equivalente deve ter bordas duplas ou múltiplas no gráfico base.

[0045] FIG. 10 é um diagrama em bloco de um dispositivo de comunicações 1000 de acordo com algumas modalidades. O dispositivo de comunicações 1000 inclui um transceptor 1010, um processador 1020, e memória 1030. O transceptor 1010 pode ser usado para a comunicação de dados para e/ou do dispositivo de comunicações 1000. Por exemplo, o transceptor 1010 pode receber e/ou transmitir bits de informação entre o dispositivo de comunicações 1000 e a CPU. A interface do codificador 1010 pode também produzir e/ou receber palavras-código de LDPC entre o dispositivo de

comunicações 1000 e outro dispositivo de comunicações em uma rede.

[0046] A memória 1030 pode incluir um armazenamento de dados 1032 que pode ser usado como um cache local para armazenar os bits de informação e/ou palavras-código recebidos. Além disso, a memória 1030 também pode incluir uma mídia de armazenamento legível por computador não-transitória (ex., um ou mais elementos de memória não voláteis, como EPROM, EEPROM, memória flash, um disco rígido, etc.) que pode armazenar os seguintes módulos de software:

- um módulo de codificação de LDPC 1034 para codificar um conjunto de bits de informação, usando um código de LDPC, para produzir uma palavra-código;
- um módulo de decodificação de LDPC 1036 para decodificar palavras-código de LDPC usando um código de LDPC.

Cada módulo de software pode incluir instruções que, quando executadas pelo processador 1020, podem fazer com que o codificador 1000 realize a função correspondente. Assim, a mídia de armazenamento legível por computador não-transitória da memória 1030 pode incluir instruções para realizar todas ou uma porção das operações descritas acima com relação às FIGS. 5-6. Deve-se observar que, enquanto os módulos 1034-1036 são ilustrados como software na memória 1030, qualquer dos módulos pode ser implementado em hardware, software, firmware, ou uma combinação dos anteriores.

[0047] O processador 1020, que é acoplado entre a interface de codificador 1010 e a memória 1030, pode ser qualquer processador adequado capaz de executar scripts de instruções de um ou mais programas de software armazenados no decodificador 1000 (ex., dentro da memória

1030). Por exemplo, o processador 1020 pode executar o módulo de codificação de LDPC 1034 e/ou o módulo de decodificação de LDPC 1036.

[0048] O módulo de codificação de LDPC 1034 pode ser executado pelo processador 1020 para codificar os bits de informação, usando o código de LDPC, para produzir uma palavra-código. Por exemplo, o processador 1020, na execução do módulo de codificação de LDPC 1034, pode realizar uma operação de codificação de LDPC nos bits de informação com base em um código de LDPC que é compartilhado pelo módulo de codificação de LDPC 1034 e um módulo de decodificação de um dispositivo de recepção correspondente. Cada palavra-código pode incluir os bits de informação originais assim como um conjunto de bits de paridade que pode ser usado para realizar verificações de paridade em e/ou recuperar os bits de informação originais. Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado (ex., com base em um levantamento quase cíclico).

[0049] O processador 1020, na execução do módulo de codificação de LDPC 1034, pode ainda perfurar um ou mais bits da palavra-código com base no código de LDPC correspondente. Por exemplo, os um ou mais bits de palavra-código perfurados podem corresponder a um ou mais nós perfurados, respectivamente, do código de LDPC. Como descrito acima, pelo menos alguns dos nós perfurados são fornecidos para eliminar várias bordas entre pares de nó no gráfico base para o código de LDPC elevado. Para algumas modalidades, os nós perfurados podem incluir um nó variável tendo um grau igual a, ou menor que, um número de nós de verificação do código de LDPC. Para outras modalidades, pelo menos um dos nós perfurados pode ser um nó variável de grau 2 (ex., usado para dividir um nó de verificação que

estaria de outro modo conectado a outro nó variável do código de LDPC por duas ou mais bordas).

[0050] O módulo de decodificação de LDPC 1036 pode ser executado pelo processador 1020 para decodificar as palavras-código de LDPC usando o código de LDPC. Para algumas modalidades, o processador 1020, na execução do módulo de decodificação de LDPC 1036, pode primeiro identificar um ou mais bits perfurados de uma palavra-código recebida com base no código de LDPC. O processador 1020 pode então realizar uma operação de decodificação de LDPC na palavra-código recebida, enquanto trata os bits de palavra-código perfurados como apagados. Por exemplo, o módulo de decodificação de LDPC 1036, como executado pelo processador 1020, pode definir os LLRs dos nós perfurados para zero na inicialização. Para algumas modalidades, o código de LDPC pode ser um código de LDPC elevado (ex., com base em um levantamento quase cíclico).

[0051] Como descrito acima, os bits de palavra-código perfurados podem corresponder aos respectivos nós perfurados do código de LDPC, sendo que pelo menos alguns dos nós perfurados são fornecidos para eliminar várias bordas entre pares de nó no gráfico base para o código de LDPC elevado. Para algumas modalidades, os nós perfurados podem incluir um nó variável tendo um grau igual a, ou menor que, um número de nós de verificação do código de LDPC. Para outras modalidades, pelo menos um dos nós perfurados pode ser um nó variável de grau 2 (ex., usado para dividir um nó de verificação que estaria de outro modo conectado a outro nó variável do código de LDPC por duas ou mais bordas).

[0052] Na especificação anterior, as presentes modalidades foram descritas com relação às modalidades exemplares específicas das mesmas. Será, no entanto,

evidente que várias modificações e alterações podem ser feitas sem se afastarem do escopo mais amplo da divulgação como estabelecido nas reivindicações anexas. A especificação e os desenhos, conseqüentemente, devem ser considerados em um sentido ilustrativo em vez de em um sentido restritivo. Por exemplo, as etapas do método ilustrado nos fluxogramas das Figs. 5-6 podem ser realizadas em outras ordens adequadas, múltiplas etapas podem ser combinadas em uma única etapa, e/ou algumas etapas podem ser omitidas.



### **REIVINDICAÇÕES**

1. Método de codificação de dados, o método caracterizado pelo fato de que compreende:

receber um conjunto de bits de informação;

realizar uma operação de codificação de verificação de paridade de baixa densidade elevada, LDPC, no conjunto de bits de informação para produzir uma palavra-código de um código de LDPC elevado correspondente elevado usando um grupo G de elevação cumulativa, onde o grupo G consiste de Z elementos  $g_0, g_1, \dots, g_{Z-1}$ , a regra de multiplicação de grupo sendo  $g_i g_j = g_k$ , onde a palavra-código elevada usando o grupo G consiste de n elementos da forma

$\sum_{i=0}^{Z-1} b_i g_i$ , onde  $(b_0, b_1, \dots, b_{Z-1})$  representa um vetor binário de modo que a palavra-código é um vetor binário de comprimento Zn, em que o código LDPC elevado correspondente é representado por uma matriz H de verificação de paridade elevada de dimensão mxn compreendendo m colunas de paridade e n-m colunas de informação e uma submatriz quadrática M é formada ao tomar as colunas de paridade m, e em que:

uma ordem do grupo de elevação, Z, corresponde a um tamanho da elevação;

em que a operação de codificação compreende inverter o determinante da submatriz quadrática M, o determinante da submatriz quadrática M sendo um polinômio

de forma:  $g_a + (g_0 + g_L)P$ , em que:

a inversão é baseada em utilizar a relação de que a inversa de  $g_a + (g_0 + g_L)P$  é igual a:

$$g_a^{-2^k} \cdot (g_a^{2^{k-1}} + (g_0^{2^{k-1}} + g_L^{2^{k-1}}) \cdot P) \cdot (g_a^{2^{k-2}} + (g_0^{2^{k-2}} + g_L^{2^{k-2}}) \cdot P) \cdot \dots \cdot (g_a + (g_0 + g_L) \cdot P),$$

$g_0$  sendo o elemento de identidade do grupo de elevação G,

$$g_0 = g_L^{2^k},$$

k sendo um inteiro positivo,

a sendo arbitrário, e

P sendo um elemento não-zero arbitrário de um anel de grupo binário associado com o grupo de elevação.

2. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que o grupo de elevação é um grupo cíclico, em que:

$g_i$  pode ser identificado com  $x^i$ ;

o determinante da submatriz quadrática esta na forma:  $x^a + (1+x^L)P(x)$ , em que  $P(x)$  possui pelo menos dois termos e  $2^k L = 0$  módulo  $Z$ .

3. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que todos os elementos abaixo da primeira subdiagonal da submatriz quadrática são iguais a 0.

4. Memória caracterizada pelo fato de que compreende instruções que, quando executadas por um processador fornecido dentro de um dispositivo de comunicações, fazem com que o dispositivo realize o método conforme definido em qualquer uma das reivindicações 1 a 3.

5. Codificador caracterizado pelo fato de que compreende:

meios para receber um conjunto de bits de informação;

meios para realizar uma operação de codificação de verificação de paridade de baixa densidade elevada, LDPC, no conjunto de bits de informação para produzir uma palavra-código de um código de LDPC elevado correspondente elevado usando um grupo G de elevação cumulativa, onde o grupo G consiste de Z elementos  $g_0, g_1, \dots, g_{Z-1}$ , a regra de

multiplicação de grupo sendo  $g_i g_j = g_k$ , onde a palavra-código elevada usando o grupo  $G$  consiste de  $n$  elementos da forma

$\sum_{i=0}^{Z-1} b_i g_i$ , onde  $(b_0, b_1, \dots, b_{Z-1})$  representa um vetor binário de modo que a palavra-código é um vetor binário de comprimento  $Zn$ , em que o código LDPC elevado correspondente é representado por uma matriz  $H$  de verificação de paridade elevada de dimensão  $m \times n$  compreendendo  $m$  colunas de paridade e  $n-m$  colunas de informação e uma submatriz quadrática  $M$  é formada ao tomar as colunas de paridade  $m$ , e em que:

uma ordem do grupo de elevação,  $Z$ , corresponde a um tamanho da elevação;

em que a operação de codificação compreende inverter o determinante da submatriz quadrática  $M$ , o determinante da submatriz quadrática  $M$  sendo um polinômio de forma:  $g_a + (g_0 + g_L)P$ , em que:

a inversão é baseada em utilizar a relação de que a inversa de  $g_a + (g_0 + g_L)P$  é igual a:

$$g_a^{-2^k} \cdot (g_a^{2^{k-1}} + (g_0^{2^{k-1}} + g_L^{2^{k-1}}) \cdot P) \cdot (g_a^{2^{k-2}} + (g_0^{2^{k-2}} + g_L^{2^{k-2}}) \cdot P) \cdot \dots \cdot (g_a + (g_0 + g_L) \cdot P),$$

$g_0$  sendo o elemento de identidade do grupo de elevação  $G$ ,

$$g_0 = g_L^{2^k},$$

$k$  sendo um inteiro positivo,

$a$  sendo arbitrário, e

$P$  sendo um elemento não-zero arbitrário de um anel de grupo binário associado com o grupo de elevação.

6. Dispositivo de comunicações (400) caracterizado pelo fato de que compreende um codificador conforme definido na reivindicação 5.

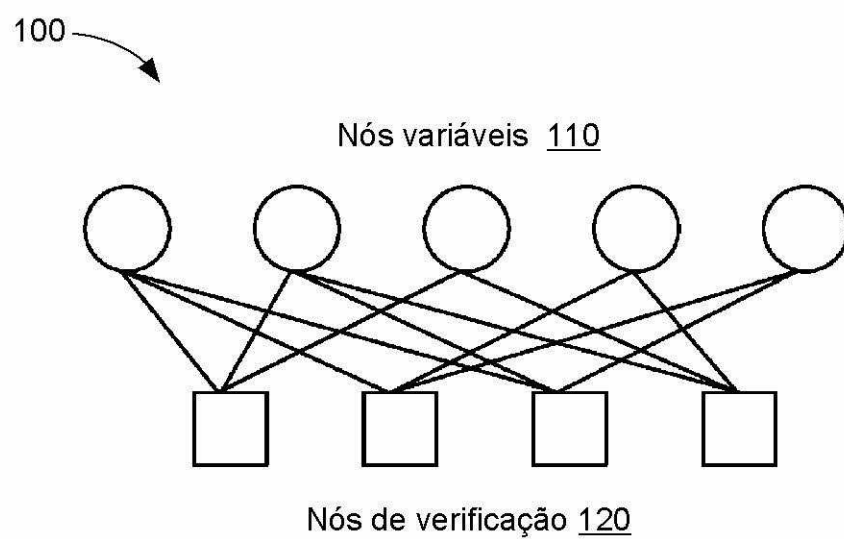


FIG. 1A

150

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

FIG. 1B

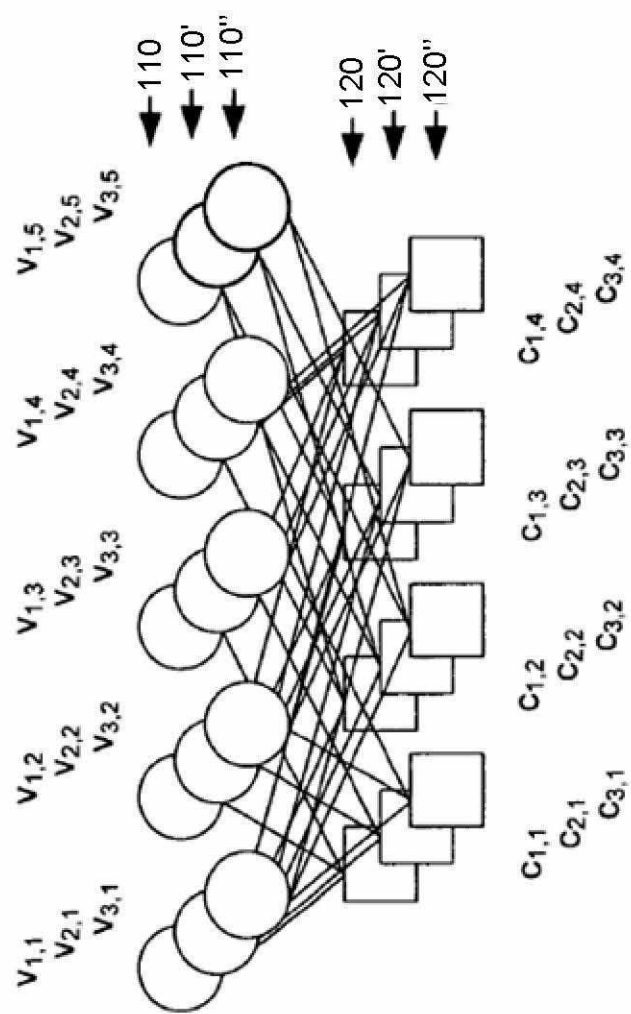


FIG. 2

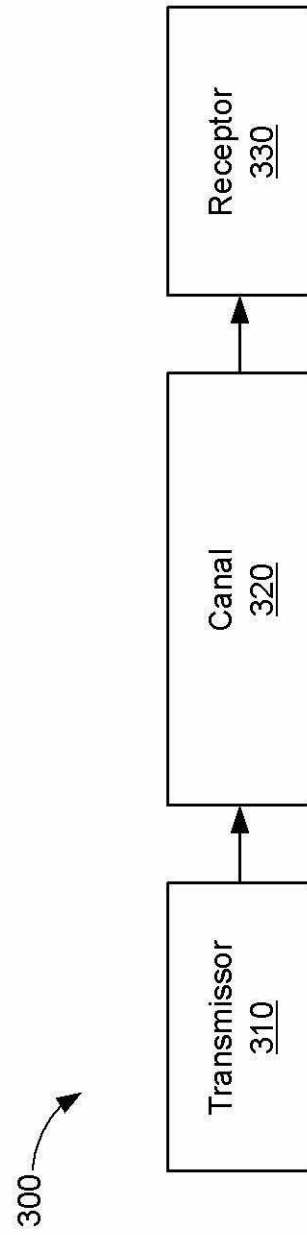


FIG. 3

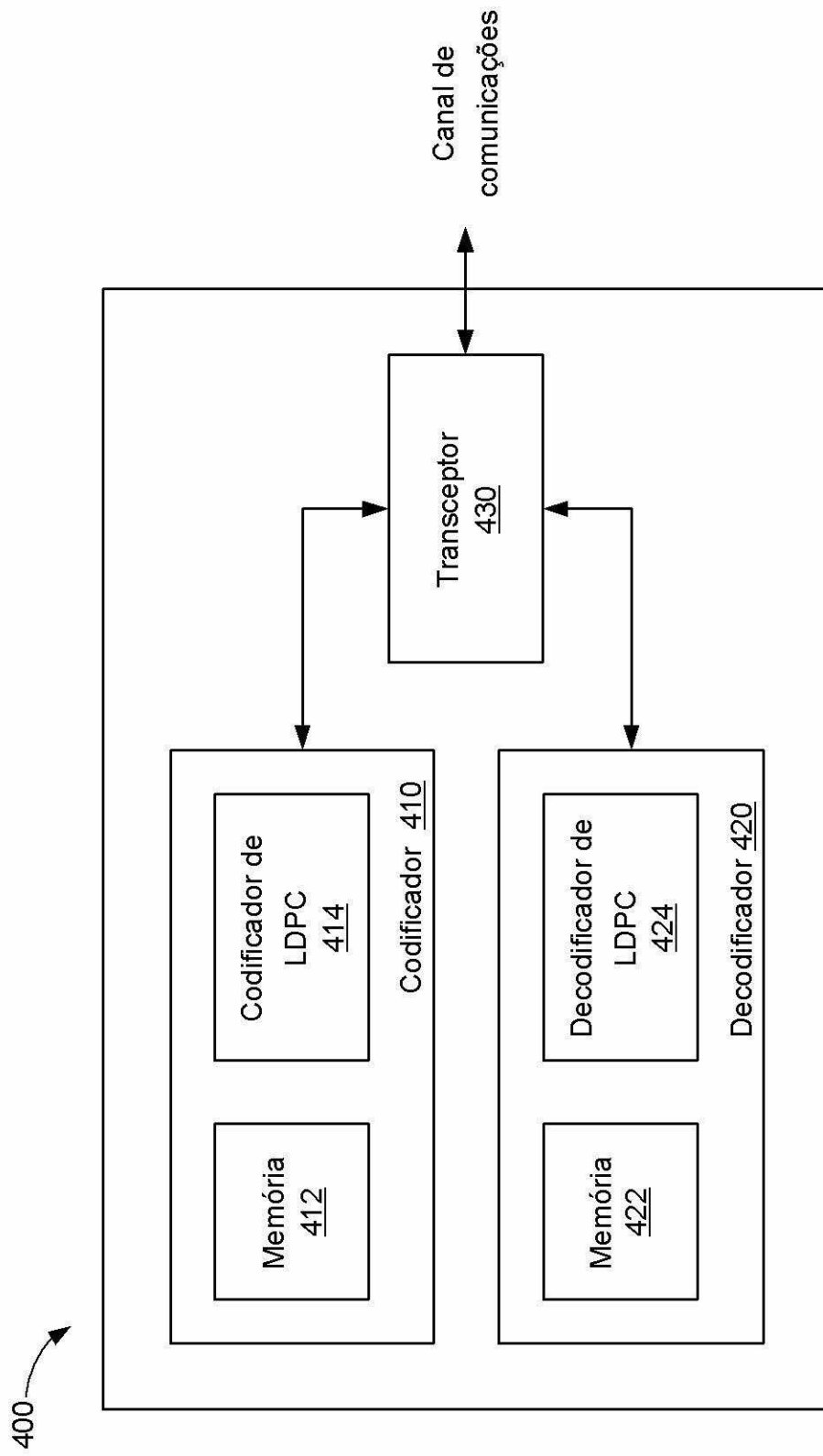
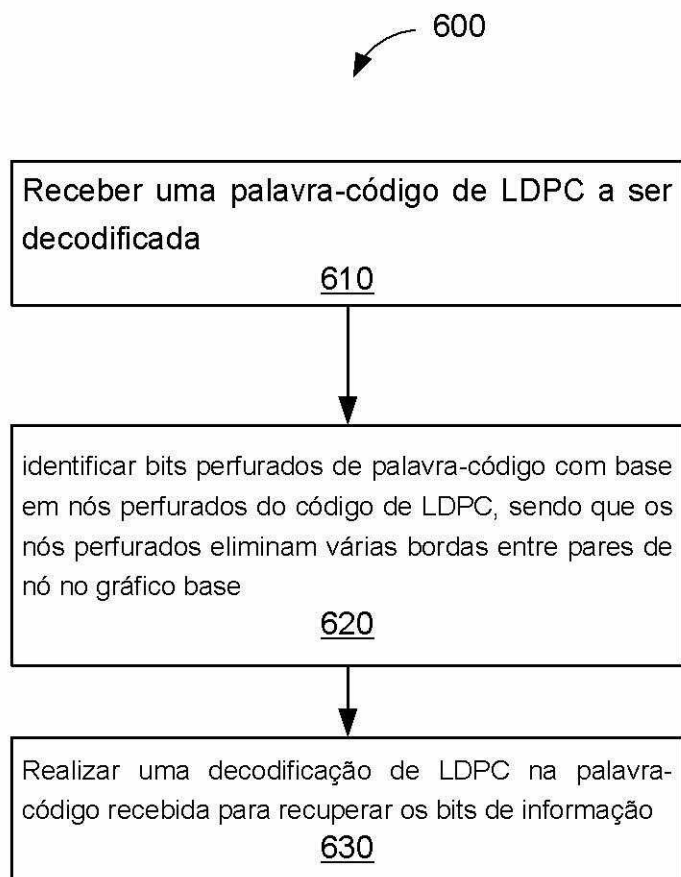


FIG. 4

**FIG. 5**



**FIG. 6**

700

70	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	18	.	13	.	.	2	.	277	112	.	233	272	85	277	104	130	355	168	43	104	105	142	67	85	277	314	238	156	20	5
0	241	236	68	187	218	.	39	209	19	269	.	.	94	323	65	64	.	46	326	188	118	122	167	344	126	281	40	85	222	.
.	241	122	0	187	66	216	354	208	.	59	111	114	277	.	64	57	.	383	182	159	175	77	256	284	92	148	185	86	184	93
.	.	122	90	29	224	168	97	.	30	254	114	334	169	132	.	.	353	141	217	105	84	138	136	31	97	3	144	20	6	354

FIG. 7

800

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	.	.	45	.	92	28	.	137	111	344	2	338	190	258	328	13
0	241	.	.	284	.	.	186	.	198	185	334	76	148	236	93	190
.	241	122	119	171	17	244	303	218	356	258	53	181	330	271	279	150
.	.	122	0	287	36	135	84	72	245	208	303	239	124	176	284	121

FIG. 8



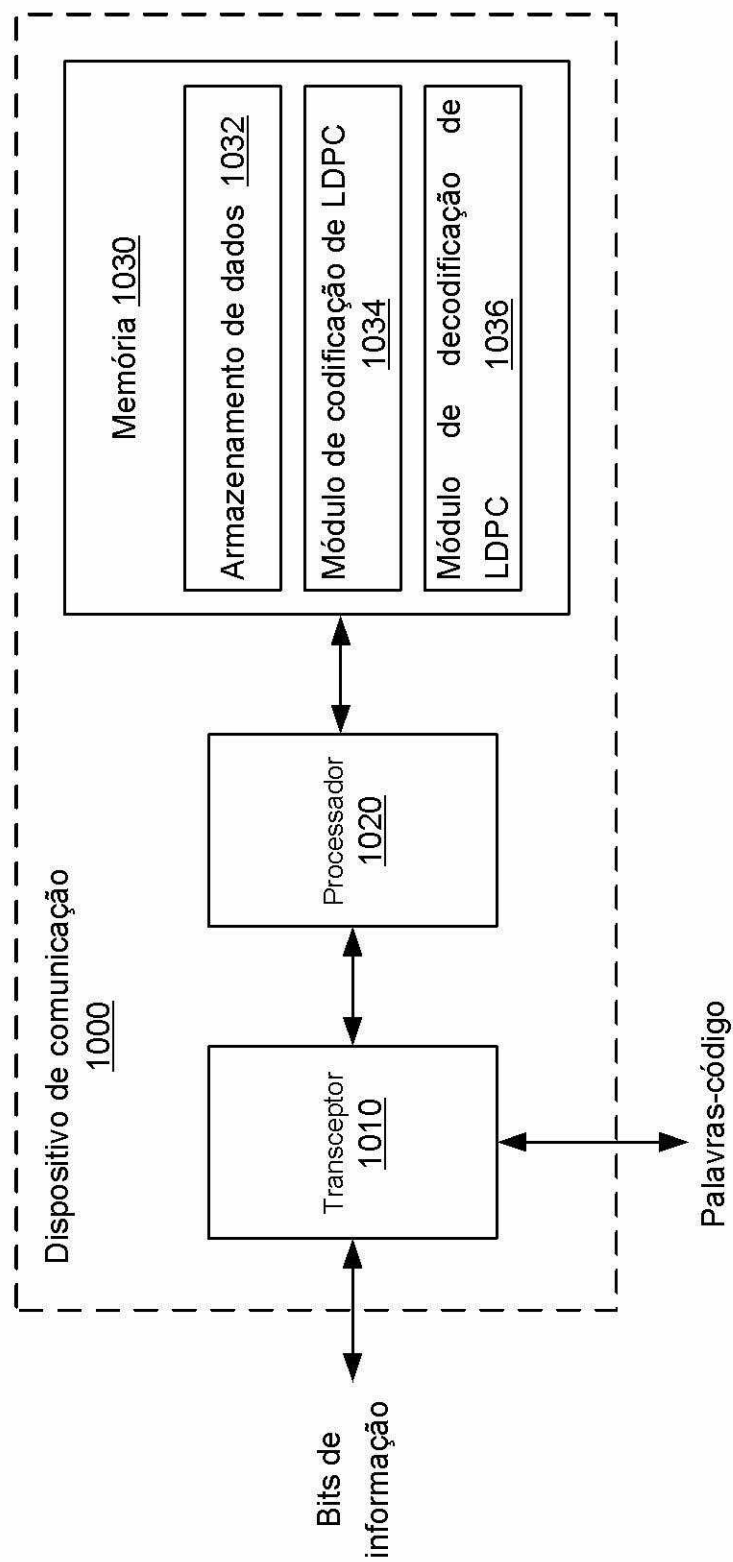


FIG. 10