



Departamento de Ciência da Computação
Universidade de Brasília

Simulador da Camada Física

Teleinformática e Redes I

Alice Borges | 18/0011855
Ana Beatriz Pontes | 18/0012428
João Pedro Silva | 17/0013944

Brasília
2022

1. Introdução

Neste projeto vamos simular de forma prática e simples como seria a implementação de uma transmissão de dados através da simulação de camadas de aplicação e uma camada física mais elaborada junto de um meio de comunicação simbólico.

Este simulador é composto pela camada de aplicação e física dividindo em camadas menores, sendo elas: Aplicação Transmissora, Camada de Aplicação Transmissora, Camada Física Transmissora, Meio de Transmissão, Camada Física Receptora, Camada de Aplicação Receptora e Aplicação Receptora.

O simulador implementado funciona da seguinte forma: o simulador mostra uma mensagem ao usuário perguntando qual mensagem ele deseja transmitir a partir da camada de aplicação, esta mensagem é convertida para bits por meio da Camada de Aplicação Transmissora e, após isso, é enviada a Camada Física Transmissora. A Camada Física Transmissora transmite o fluxo de bits para o Meio de Comunicação, onde os bits são transferidos para a Camada Física Receptora, onde por fim a mensagem é enviada para a Camada de Aplicação Receptora e é decodificada e mostrada da forma correta.

Foram utilizados três tipos de codificação: Codificação Binária, Codificação Manchester, Codificação Bipolar.

O comportamento da codificação Binária, Manchester e Bipolar estão descritas nas Figuras 1 e 2. A explicação e implementação dessas será detalhada na Seção 2.

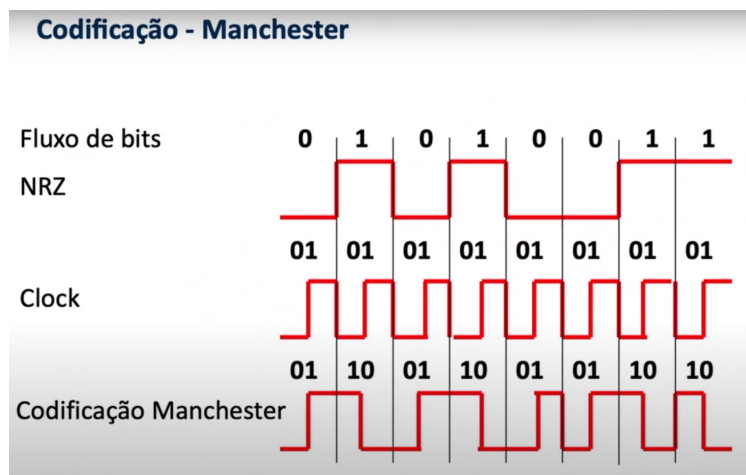


Figura 1 - Codificação Binária e Manchester

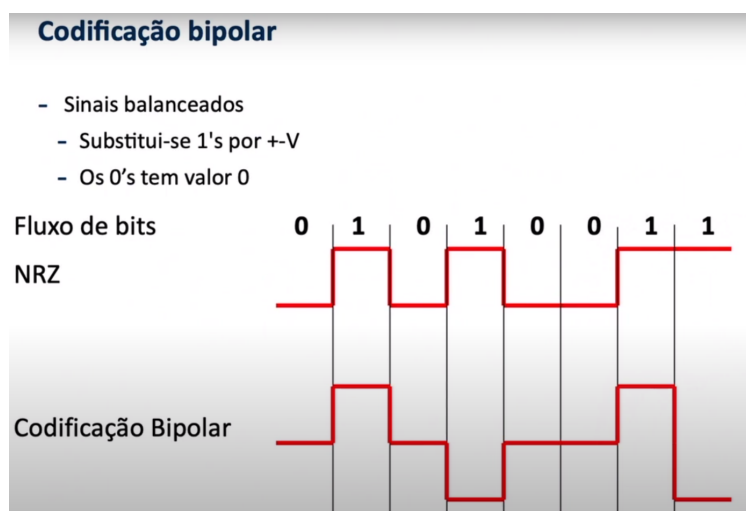


Figura 2- Codificação Binária e Bipolar

Na Seção 2 será descrita a implementação detalhadamente do simulador de camada física e apresentado os resultados da execução. Já a Seção 3 apresenta os membros da equipe as atividades realizadas por cada um. Por fim, a Seção 4 apresenta uma conclusão sobre o trabalho e descreve dificuldades e limitações enfrentadas neste trabalho.

O código desse trabalho pode ser encontrado no github a partir do link:

<https://github.com/aliceborges42/CamadaFisica/tree/main> .

2. Implementação

Para a implementação do simulador da camada física, nos baseamos no diagrama da Figura 3, onde cada quadrado representa um protocolo e cada protocolo será implementado por meio de uma sub-rotina, dessa forma a Figura 4 apresenta um diagrama com os detalhes do funcionamento de cada função escrita no código que representam tais protocolos.

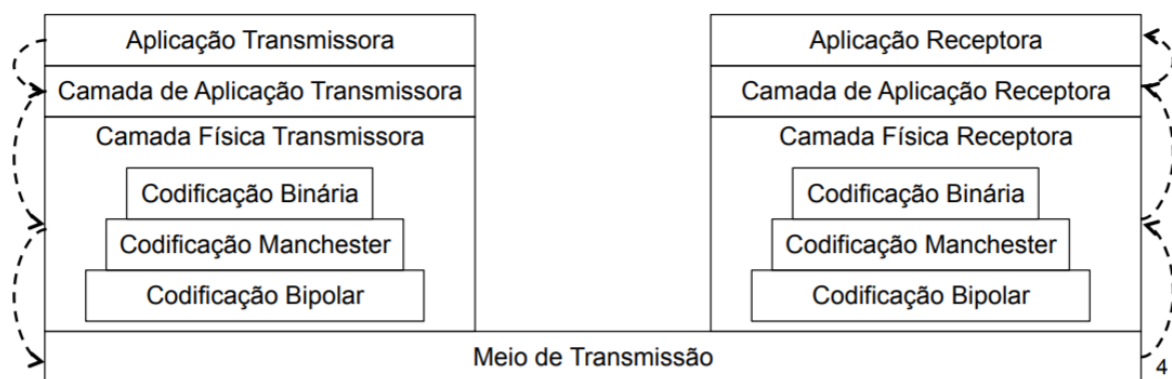


Figura 3 - Diagrama de representação da camada física e aplicação

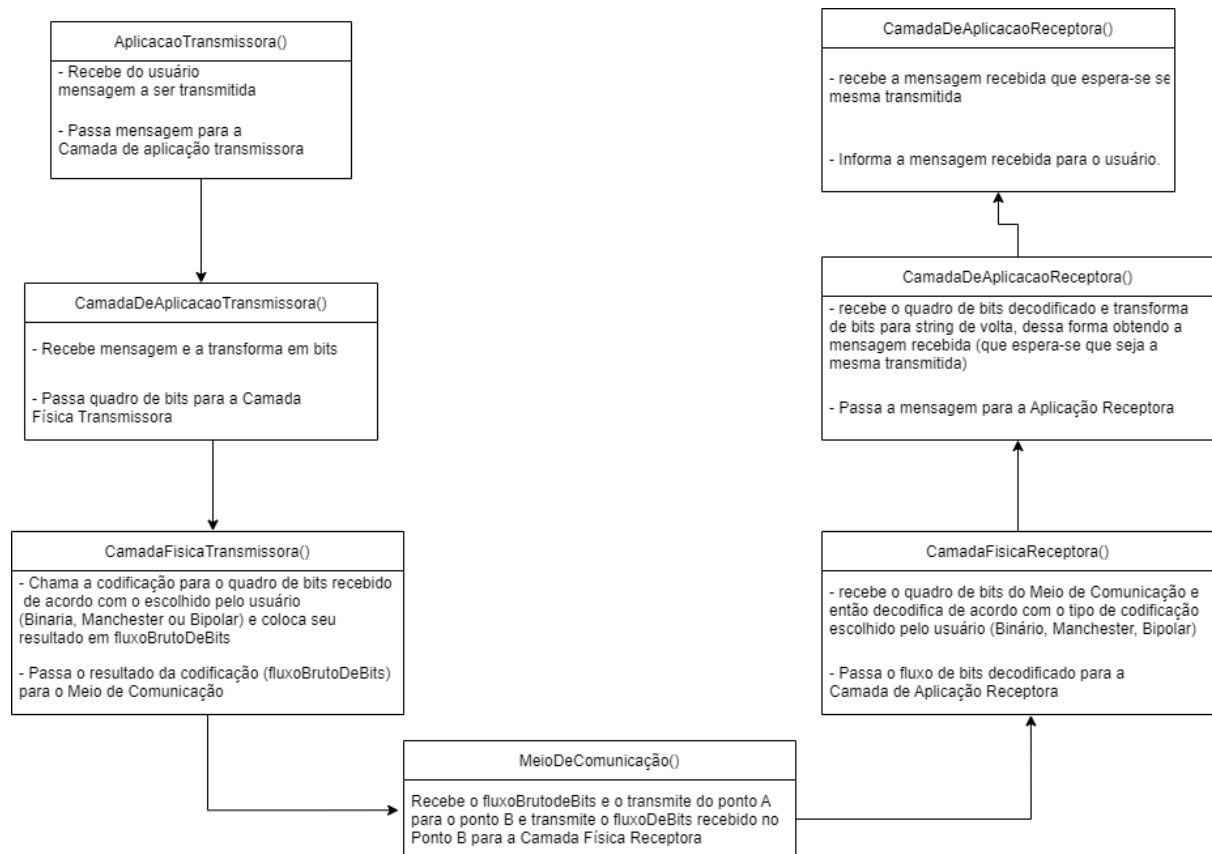


Figura 4 - Diagrama com explicações do funcionamento do código

A função Aplicação Transmissora é a primeira função a ser executada, na qual recebe a mensagem que o usuário quer transmitir, em seguida essa mensagem é enviada para a próxima função, a função Camada de Aplicação. Nessa função a mensagem recebida é transformada em bits conforme explicado em **2.1**. Após isso, é repassado o quadro de bits transformado para a Camada Física Transmissora. Na função Camada Física Transmissora é realizada a codificação escolhida pelo usuário, nas quais podem ser a Binária, Manchester ou Bipolar, tais codificações são explicadas em **2.4**, então após a codificação ser realizada, é retornado um fluxo bruto de bits que é direcionado para o Meio de Comunicação, que tem como função estabelecer a comunicação entre as camadas físicas de transmissão e recepção, como é explicado no diagrama da Figura 4, então é transmitido para a Camada Física Receptora os dados codificados. A função Camada Física Receptora recebe esses dados então os decodifica de acordo, novamente, com o tipo de codificação escolhido, e então os dados decodificados são passados para a próxima camada que seria a Camada de Aplicação Receptora. Em Camada de Aplicação Receptora é realizado a transformação dos bits decodificados que estão em binário para string, como explicado em **2.2**, e o resultado da transformação é passado enfim para a última camada, a Aplicação Receptora que então entrega ao usuário a mensagem que foi transmitida e então recebida.

A implementação contou com a criação de uma GUI com a ajuda da biblioteca [ncurses](#), em **2.3** é detalhado a forma com que essa parte foi implementada.

Por fim, em **2.5** é apresentado os resultados da execução do código, utilizando as três codificações, como é possível ver nas Figuras, a mensagem foi transmitida e recebida corretamente, como esperado.

2.1 - Transformação da mensagem em bits

Para transformar a string mensagem em bits e assim trabalhar a codificação dessa, utilizamos de um biblioteca do C++ chamada [bitset](#), na qual teve a função de transformar cada letra da mensagem em 8 bits binários a partir dessa transformação colocamos cada bit em um [vector](#), outra biblioteca utilizada que nos permitiu fácil manipulação da estrutura de dados. Então para cada letra da mensagem houve sua transformação em binário (8 bits) e cada bit foi colocado num vector de inteiro, dessa forma obtemos o quadro de bits utilizado na codificação e transmissão.

2.2 - Decodificação de bits para mensagem

Para decodificar os bits e convertê-los de volta para string, esse processo começa fazendo a leitura do vector onde está armazenado os bits, durante essa leitura vamos processar grupos de 8 bits seguidos até o final do vector, com base nos bit que estão setados em 1 fazemos a conversão de binário para decimal usando uma variável acumuladora, que vai somar sempre que um bit for 1. Utilizando a posição do bit setado no grupo de 8 bits como expoente e iremos fazer uma potencia simples de base dois, após terminar o processamento do grupo de 8 bits, convertemos o valor inteiro para um caractere utilizando um type casting para char e depois concatenamos esse caractere na string que será apresentada ao final da execução, ao final da concatenação repetimos todo o ciclo de processamento até percorrer todo o vector e retorna a mensagem decodificada e em formato string.

2.3 - Ncuses

Para a interface e comunicação com o usuário optamos por utilizar o Ncurses, com ele foi possível montar uma interface onde o projeto é executado de maneira a parte do terminal e agrupar as interações de maneira mais visível e organizada.

Foram utilizadas funções básicas de início de tela(`initscr`) e encerramento de tela(`endwin`), apresentação de conteúdo na tela (`printw`), funções de input (`scanw`) e funções de manipulação de cores (`start_color`, `init_pair`, `attron`, `attroff`).

2.4 - Codificações

Os protocolos de codificação utilizados serão apresentados a seguir:

- Codificação Binária: Esse protocolo foi implementado utilizando a técnica de codificação NRZ-L(Non Return Zero - Level), essa técnica é representada pela presença de tensão, que atribuímos ao sinal 1 lógico, e pela ausência de tensão no pulso transmissor relacionada ao sinal lógico 0. A implementação foi feita de forma simples apenas transmitindo os bits da mensagem pela camada física de transmissão sem sofrer nenhuma mudança devido ao

padrão de codificação e depois de recebido na camada física de recepção, os bits que chegam não precisam ser decodificados e são repassados para a camada de aplicação receptora, onde são transformados de bits para uma string e logo em seguida são apresentados ao usuário.

- **Codificação Manchester:** Esse protocolo foi implementado utilizando a técnica de codificação Manchester, a mesma é representada pela capacidade de manter receptor e transmissor sincronizados por meio de um sinal pulsado que combina tanto o sinal de dados quanto o sinal de clock. A implementação foi feita codificando os bits que chegam da camada de aplicação transmissora, essa codificação ocorre usando estruturas condicionais para representar uma porta xor e utilizando o bit atual e os valores de 0 e 1 de subida do clock como entradas, gerando assim uma saída de dois bits que combina o bit atual e o clock. Após a codificação os bits saem da camada física transmissora e são encaminhados para a camada física receptora pela camada de meio de transmissão, ao chegar na camada receptora os bits são separados do clock, por meio de uma decodificação simples, também usando estruturas condicionais, e depois seguem para camada de aplicação receptora e para serem convertidos em string para ser apresentados ao usuário.
- **Codificação Bipolar:** Esse protocolo foi implementado utilizando a técnica de codificação Bipolar, nessa codificação serão utilizados três níveis de para representar os símbolos binários, a ausência de sinal representa o nível lógico 0 e os sinais que representa a presença de pulsos positivos e negativos são associados ao nível lógico 1. A implementação foi feita codificando os bits que chegam da camada de aplicação transmissora, essa codificação ocorre usando estruturas condicionais para manter o valor do bit quando for zero e caso seja um, vamos gerar um sinal de valor $+V$ ou $-V$, para isso utilizamos um flag que checagem para saber se o último sinal codificado que possuía valor era $+V$ ou $-V$. Em seguida os bits seguem pela saída da camada física transmissora e vão para a camada física receptora sendo conduzidos por meio da camada meio de transmissão, ao chegar na camada receptora os bits decodificados por meio de uma checagem simples, também usando estruturas condicionais, mantendo os bits de sinal 0 e convertendo todo sinal, seja ele $+V$ ou $-V$ para um bit de nível lógico 1, concluída essa parte, os bits seguem para camada de aplicação receptora e para serem convertidos em string para ser apresentados ao usuário.

2.5 - Exemplos de resultados do simulador para cada codificação:

Logo abaixo estão os resultados da execução do código para cada codificação, como mensagem utilizamos “Simulador da Camada Física”.

Codificação Binária

```

Aplicacao Transmissora
Digite uma mensagem: Simulador da Camada Fisica

Escolha o tipo de codificação
0 - Binaria
1 - Manchester
2 - Blpolar

0

Mensagem a ser transmitida: Simulador da Camada Fisica
Codificação escolhida: Binaria

Camada De Aplicacao Transmissora
0101001101101001011011010111010101100011000010110010001101111011100100010000001100100011000010010000010000110110000101101101010000101001
00011000010010000001000110110000111010110101110011011010010110001101100001

Camada Fisica Transmissora - Codificação Binaria:
01010011011010010110110101110101011011000110000101100100011011110111001000100000011001000110000100100000100001101100001011011010100001011001
000110000100100000010001101100001110101101011100110110100101100010110001101100001

Camada Fisica Receptora - Codificação Binaria:
010100110110100101101101011101010110110001100001011001000110111101111001000100000011001000110000100100000100001101100001011011010100001011001
00011000010010000001000110110000111010110101110011011010010110001101100001

Camada Fisica Receptora
Sua mensagem foi recebida

Camada de Aplicação Receptora
A mensagem recebida foi: Simulador da Camada Fisica

```

Codificação Manchester

[illegible]

Codificação Bipolar

```

Aplicação Transmissora
Digite uma mensagem: Simulador da Camada Física

Escolha o tipo de codificação
0 - Binária
1 - Manchester
2 - Bipolar

2

Mensagem a ser transmitida: Simulador da Camada Física
Codificação escolhida: Bipolar

Camada De Aplicacao Transmissora
010100110110100101101101011011000110000101100100011011101110010001000000100100011000010010000001000011011000010110110101100001011001
0001100001001100000010001101100001110101101011100110110011011010011011000110110001

Camada Fisica Transmissora - Codificação Bipolar:
010-1001-101-10100-101-101-1010-11-1010-101-101-10001-1000010-1100-10001-101-11-101-1100-10001000000-1100-10001-10000100-100000010000-110-1100
00-101-101-1010-10000-101-1001000-110000-1001000000-10001-101-100001-110-101-1010-11-1001-101-10100-101-10001-101-100001

Camada Fisica Receptora - Codificação Bipolar:
010100110110100101101101011101010110110001100001011001000110111011100100010000001100100011000010010000001000011011000010110110101100001011001
000110000100110000001000110110000111010110101110011011010010110001101100001

Camada Fisica Receptora
Sua mensagem foi recebida

Camada de Aplicação Receptora
A mensagem recebida foi: Simulador da Camada Física

```

3. Membros e Atividades

- **Alice Borges:** implementação da transformação da string para bits e bits para string, codificação e decodificação binária, manchester e bipolar, elaboração do relatório.
- **Ana Beatriz Pontes:** implementação da codificação e decodificação Bipolar, elaboração do relatório, interface GUI com ncurses/interação com o usuário.
- **João Pedro:** implementação da transformação da string para bits e bits para string, codificação e decodificação binária, manchester e bipolar, elaboração do relatório.

4. Conclusão

Chegando ao término desse projeto, podemos concluir que foi interessante a abordagem prática sobre o conteúdo teórico da disciplina. Ao longo da construção do trabalho, pesquisas em [bibliografias adicionais](#) foram feitas para que os conhecimentos sobre as codificações fosse reafirmados e assim obtendo melhor entendimento para desenvolver o código fonte.

Algumas dificuldades ao longo do projeto devem ser ressaltadas, podemos começar com a carga de tarefas e trabalhos devido ao final do semestre acabou atrasando a entrega do projeto, além disso tivemos algumas dificuldades de compreender e codificar a parte de interface gráfica, porém conseguimos concluir o projeto com um resultado, cremos que, satisfatório. Outra dificuldade enfrentada foi de fato entender como implementar a teoria das codificações em código C++, porém com o trabalho em grupo e discussões proveitosas foi possível chegar a um consenso e termos resultados práticos na codificação.

Por fim, podemos também enfatizar sobre os resultados obtidos que acreditamos que sejam satisfatórios e que sejam o esperado de acordo com a especificação do projeto.