

NULL MODEM LINUX - WINDOWS

Maria Antônia Espinosa Woltmann
Curso de Ciência da Computação
Universidade Franciscana
CEP 97010-032 – Santa Maria, RS, Brasil
maria.aewoltmann@ufn.edu.br

Luiza Lewandowski Karlec
Curso de Ciência da Computação
Universidade Franciscana
CEP 97010-032 – Santa Maria, RS, Brasil
luiza.karlec@ufn.edu.br

Vanessa Cezar do Nascimento
Curso de Ciência da Computação
Universidade Franciscana
CEP 97010-032 – Santa Maria, RS, Brasil
vanessa.cezar@ufn.edu.br

Sylvio André Garcia Vieira
Curso de Ciência da Computação
Universidade Franciscana
CEP 97010-032 – Santa Maria, RS, Brasil
sylvio@ufn.edu.br

I. INTRODUÇÃO

Ao considerarmos as formas existentes de comunicação de dados na indústria atual, é possível verificarmos que existem padrões clássicos, como por meio de cabos coaxiais, par trançado, fibra ótica. Dessa maneira, visando a aprendizagem acerca do funcionamento prático das formas de comunicação de dados e focando na aprendizagem acerca comunicação serial, foi realizado em laboratório estudos sobre de seu funcionamento.

Nesta direção, considerando a relevância da comunicação serial na atualidade, como em projetos de hardware modernos, demonstraremos como foi realizado a conexão e transmissão de dados por meio desse modelo de comunicação, utilizando diferentes máquinas com diferentes sistemas operacionais. O que é possível com o uso do cabo correto, que possui compatibilidade entre diferentes fabricantes para realizar a interconexão.

A. Objetivo Geral

O objetivo geral deste trabalho é desenvolver um relatório sobre como foi promovido, em sala de aula, a comunicação serial via cabo null modem entre dois computadores com Sistemas Operacionais distintos, um utilizando Windows e outro utilizando Linux, os dois computadores foram conectados via cabo com conectores DB9. Em que a tarefa do trabalho consiste em promover a passagem de um trem entre as telas dos dois computadores.

B. Objetivos Específicos

- Compreender sobre comunicação serial e qual a pinagem correta para interligar dois computadores pela interface serial.
- Praticar soldagem para conseguir realizar a ligação entre os fios do cabo e os conectores DB9.
- Utilizar na máquina com Sistema Operacional do Windows o emulador de terminal VT100 TeraTerm.

- Compreender a configuração e como habilitar a porta de comunicação serial no Sistema Operacional Linux - Ubuntu.
- Promover a comunicação entre os dois computadores via cabo, em que o computador Linux deve ser acessado por meio do Windows.

C. Materiais utilizados neste trabalho

- Cabo de rede
- Conector DB9 fêmea
- Alicates desencapador
- Solda
- 2 computadores, um com sistema operacional Windows e outro Linux
- Software de terminal serial Tera Term utilizado no Windows

II. METODOLOGIA

A. Comunicação Serial via cabo null modem

A comunicação serial é um método de transmissão de dados em que os bits são enviados sequencialmente, um de cada vez, através de um único canal de comunicação. Sua simplicidade, custo-efetividade e capacidade de conectar múltiplos dispositivos em rede o tornam uma escolha popular em diversas aplicações. Para que determinados dispositivos funcionem e passem a receber e enviar os dados, é necessário, na maioria dos casos, o uso de portas seriais. [1]

As portas seriais são utilizadas para conectar fisicamente dispositivos assíncronos a um computador, elas geralmente estão localizadas na parte de trás da unidade do sistema, seja integrado ou utilizando um adaptador multiport. Essas portas seriais requerem apenas um único pino, ou fio, para enviar um caractere de dados para o dispositivo. Para realizar isso, os dados são convertidos a partir de uma forma paralela (enviada pelo computador), para uma forma sequencial, em que bits são organizados um após o outro em uma série. Os dados

são então transmitidos para o dispositivo com o bit menos significativo (ou zero-bit) enviado primeiro. Depois que os dados são recebidos pelo dispositivo remoto, os dados são convertidos de volta em forma paralela.[2]

O padrão de comunicação serial utilizado neste trabalho foi o RS-232C que tem por objetivo padronizar um método de interconexão entre terminais e canais de comunicação de dados. Ele define um modo de troca de sinais de controle (protocolo) e de sinais de dados serializados entre o terminal e o canal de comunicação de dados. Além disso, especifica características mecânicas, como conectores, a serem utilizados, servindo de referência para projetistas de equipamentos.[3] Nesse trabalho foi utilizado o conector DB9 que será detalhado mais a frente.

Durante a realização deste trabalho, foi empregada a comunicação serial via cabo Null Modem, ou seja, estabeleceu-se uma conexão direta entre dois computadores utilizando apenas um cabo, sem a necessidade de modems ou roteadores intermediários. Para que isso fosse possível, foi necessário cruzar os fios de transmissão e recepção dentro do cabo. Essa forma de conexão, que permite a troca direta de dados entre os dispositivos, será detalhada mais adiante

B. Cabo de Rede

Para realizar a ligação entre os dois computadores, um com sistema operacional Windows e outro com Linux, foi utilizado um cabo de par trançado cat5 adaptado com conectores DB9, permitindo a comunicação serial entre os dispositivos.

Cabo de par trançado é um tipo de cabo de rede formado por oito fios internos, organizados em quatro pares trançados, sendo cada fio com uma cor. No caso do cabo utilizado para fazer o trabalho ele é composto por fios das cores: azul claro, azul escuro, marrom claro, marrom escuro, verde claro, verde escuro, laranja e branco. Cada par de fio trançados criam seu próprio campo elétrico, que ajuda a reduzir a interferência eletromagnética entre eles, possibilitando assim um maior fluxo e uma melhor performance da transmissão de dados.

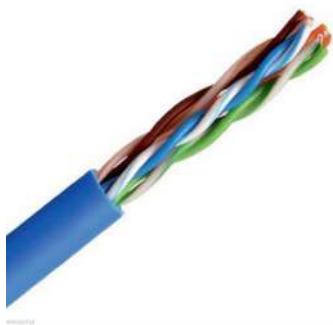


Figura 1. Cabo de par trançado cat5

Abaixo estão fotos tiradas durante a realização do trabalho, no qual foi necessário decapar o cabo de par trançado cat5, fornecido pelo professor, para isso utilizou-se um alicate desencapador. Em seguida, foram escolhidos três fios, verde claro, laranja e branco, para serem conectados ao conector

DB9, esses fios também foram decapados com auxílio do alicate [4].



Figura 2. Utilizando um alicate para decapar o cabo



Figura 3. Utilizando um alicate para decapar os fios individualmente

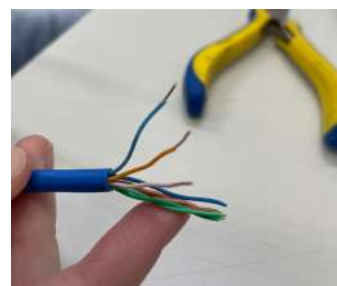


Figura 4. Os três fios que serão utilizados na ligação já decapados

C. Conector DB9

Os dispositivos de comunicação serial utilizam conectores do tipo D de 9 pinos para conexões cabeadas.[5] O conector DB9, um tipo de conector D-Subminiature (D-Sub), é amplamente empregado em comunicações seriais RS-232. Seu formato em "D" refere-se à carcaça metálica com essa forma, que protege os pinos, reduz interferências eletromagnéticas e assegura a orientação correta durante a conexão.

Esse conector pode ser encontrado nas versões macho (com pinos) ou fêmea (com furos) e é amplamente utilizado para a conexão entre computadores e dispositivos periféricos, como modems, impressoras, roteadores industriais e equipamentos de automação. Neste trabalho foi utilizado apenas o DB9 fêmea.



Figura 5. Conector DB9 fêmea

Porta COM (Communication Port) é a designação lógica utilizada pelos sistemas operacionais para identificar interfaces de comunicação serial, essas portas permitem a transmissão de dados de forma sequencial. A interface física mais comum para essas conexões é o conector DB9, que serve como meio de acesso à porta COM. Dessa forma, ao conectar um dispositivo através de um cabo com conector DB9, o sistema o reconhece e estabelece a comunicação por meio da porta COM correspondente, facilitando a troca de dados via protocolo RS-232.

Cada pinagem do conector tem sua própria função distinta como visto na imagem abaixo.

Pino	Sinal	Descrição do sinal
1	DCD	Data Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	Signal Ground (Terra)
6	DSR	Data Set Ready
7	RTS	Request to Send
8	CTS	Clear to Send
9	RI	Ring Indicator

Figura 6. Pinagem do conector DB9

A comunicação serial por meio do conector DB9 ocorre através da troca de sinais entre os pinos, seguindo o protocolo RS-232. Quando os dados estão prontos para serem transmitidos, o dispositivo transmissor sinaliza essa condição utilizando os pinos de controle.

Os pinos 1 (DCD - Data Carrier Detect) e 6 (DSR - Data Set Ready) recebem sinais oriundos do pino 4 (DTR - Data Terminal Ready), indicando que o canal está pronto e sincronizado para a transmissão de dados. Isso permite que o dispositivo reconheça que há um canal disponível e preparado para a comunicação.

Durante a comunicação propriamente dita, o dispositivo transmissor utiliza o pino 7 (RTS - Request To Send) para solicitar permissão de envio. Essa solicitação chega ao receptor através do pino 8 (CTS - Clear To Send), que responde autorizando ou não a transmissão. Em alguns cenários, é comum utilizar um jumper para interligar diretamente os pinos 7 e 8 no mesmo conector, simulando a autorização contínua de envio e eliminando o controle de fluxo por hardware. Da mesma forma, o pino 4 pode ser ligado aos pinos 6 e 1

no mesmo lado, mantendo o canal sempre ativo. Contudo, esse método elimina a capacidade de detecção de erros via hardware, deixando o canal permanentemente pronto para transmitir, independentemente do estado real da conexão.

O pino 5 é responsável pelo aterramento comum dos sinais. Já os pinos 2 (RXD - Receive Data) e 3 (TXD - Transmit Data) são os responsáveis pela transmissão e recepção dos dados propriamente ditos. Quando apenas esses pinos (2, 3 e 5) estão conectados, a comunicação depende exclusivamente de detecção de erros por software, sendo necessário o uso de protocolos em camadas superiores — como a camada 4 do modelo TCP/IP, que exige confirmação da recepção de pacotes (ACK).

Por outro lado, ao conectar todos os pinos de controle, a comunicação passa a contar com detecção de erros por hardware, pois os sinais nos pinos específicos permitem identificar falhas, como rompimento de cabo ou ausência de dispositivo no outro extremo.

Neste trabalho foi utilizado um cabo serial direto, com conectores DB9 do tipo fêmea em ambas as extremidades. A ligação seguiu o padrão de conexão um para um, em que os pinos de mesma função são conectados diretamente entre si. O pino de transmissão (TX - pino 3) de um conector é ligado ao pino de recepção (RX - pino 2) do outro, e vice-versa[5], permitindo a comunicação bidirecional. O pino 5 (GND), responsável pelo aterramento elétrico, também é interligado entre os conectores, garantindo um referencial comum de sinal.

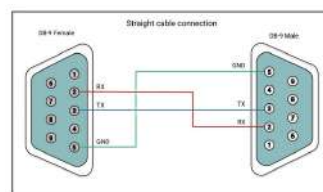


Figura 7. Exemplo de como foi feita a ligação nos conectores DB9

Abaixo estão fotos tiradas durante a realização do trabalho, em que fora soldados cada um dos três fios no conector DB9.

- Fio verde claro foi soldado no pino 2
- Fio laranja foi soldado no pino 3
- Fio branco foi soldado no pino 5

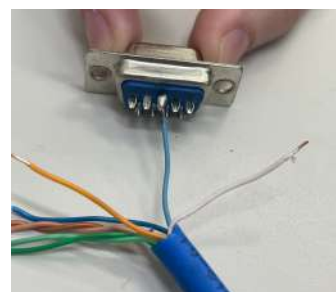


Figura 8. Um dos fios soldado ao conector DB9

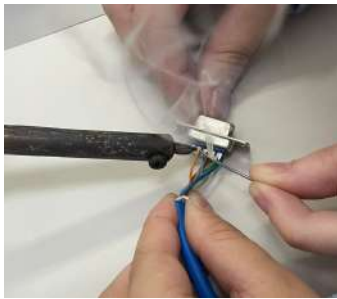


Figura 9. Soldagem dos fios no conector DB9



Figura 10. Todos os fios que seram utilizados soldados no cabo DB9

D. Comunicação entre os dois computadores

1) *Windows*: Para que fosse possível realizar a transmissão de dados de forma serial pela máquina com o Sistema Operacional Windows, foi necessária a instalação do aplicativo TERA TERM, especificamente a versão teraterm-5.0.exe, a qual foi conectada à máquina com o Sistema Operacional Linux por meio dos comandos inseridos no terminal Linux.

2) *Linux*: Já para ser possível que a máquina com o Sistema Operacional Linux se conectasse à máquina com Sistema Operacional Windows, foi necessário inserir uma série de comandos para realizar a permissão de conexão do terminal e a identificação do cabo ethernet cat5 que está conectado em ambas às máquinas.

- `sudo`: Significa "SuperUser DO". Permite executar comandos como superusuário (root), com permissões de administrador.
- `whoami`: Mostra o nome do usuário atualmente logado.
- `dmesg | grep tty`: `dmesg` mostra mensagens do kernel (núcleo no sistema operacional) no Linux, quando algo é conectado no computador, o kernel detecta e registra uma mensagem. `grep tty` filtra apenas as que contêm a palavra `tty`, onde `tty` representa qualquer porta serial.
- `chmod 777 /dev/ttyS4`: Da permissões totalmente abertas (leitura, escrita e execução) para qualquer usuário sobre a porta serial `ttyS4`, que representa a porta serial à qual o cabo ethernet cat5 está conectado.
- `ls -l /dev/ttyS*`: Lista detalhadamente todas as portas seriais que começam com `ttyS`, usado no trabalho para verificar as permissões da porta serial `ttyS4`.
- `sl > /dev/ttyS4`; `sl`: `sl` mostra um trenzinho animado passando pela tela do terminal, `> /dev/ttyS4` direciona a saída para a porta serial. `sl` novamente para mostrar o

trem no próprio terminal. Causando a impressão de que ele está passando de uma tela para a outra.

Abaixo estão fotos que foram tiradas da tela do computador com sistema operacional Linux.

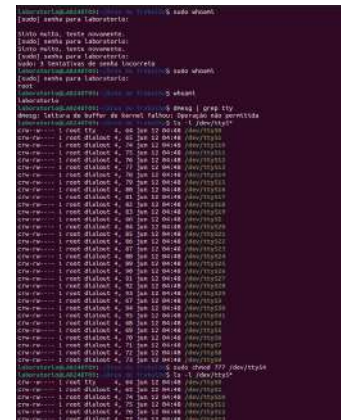


Figura 11. Tela 1 do Linux

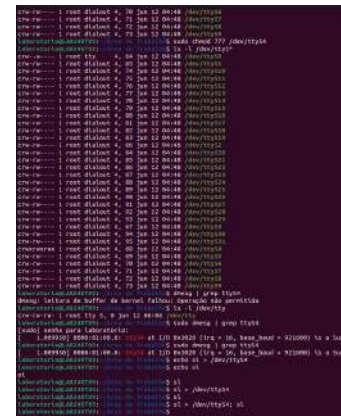


Figura 12. Tela 2 do Linux

E. Trem passando entre as duas telas

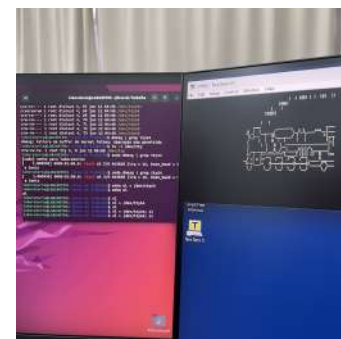


Figura 13. Início da passagem do trem na tela do Windows

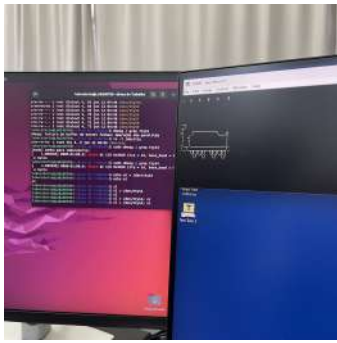


Figura 14. Fim da passagem do trem na tela do Windows

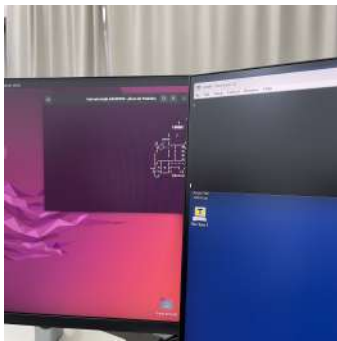


Figura 15. Início da passagem do trem na tela do Linux

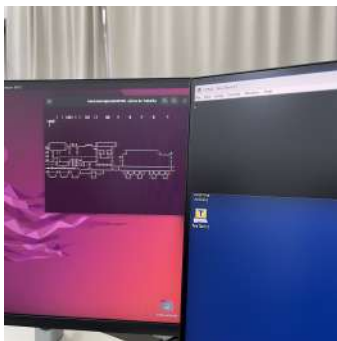


Figura 16. Trem passando na tela do Linux

III. CONSIDERAÇÕES FINAIS

Desse modo, foi possível concluir que a comunicação serial entre dois computadores com sistemas operacionais distintos é algo fácil e viável de ser realizado, sobretudo de ser estudado quando a conexão entre as máquinas não percorrem grandes distâncias.

Portanto, foi exposto que o desenvolvimento prático desse tipo de transmissão de dados é algo que promove o conhecimento sobre as possibilidades existentes de conexões atualmente.

REFERÊNCIAS

- [1] Dakol Indústria e Comércio Ltda. *Comunicação serial*. Acessado em: 14 de junho de 2025. 2025. URL: <https://www.dakol.com.br/comunicacao-serial/>.
- [2] IBM Documentation. *Comunicação serial*. Acessado em: 14 de junho de 2025. 2024. URL: <https://www.ibm.com/docs/pt-br/aix/7.3.0?topic=communications-serial-communication>.
- [3] EPUSP — Laboratório Digital. *Conceitos de Comunicação Serial Assíncrona*. Acessado em: 14 de junho de 2025. 2015. URL: https://www2.pcs.usp.br/~labdig/pdf/files_2015/conceitos-comunicacao-serial.pdf.
- [4] Sylvio Andre Garcia Vieira. *Como fazer um cabo null modem db9 db9*. Acessado em: 12 de junho de 2025. 2024. URL: <https://www.youtube.com/watch?v=RpNdDMqqLyk&t=176s>.
- [5] Olga Weis. *Compreendendo o protocolo RS232: comunicação serial e pinagem de conectores*. Acessado em: 14 de junho de 2025. 2024. URL: <https://www.virtual-serial-port.org/pt/article/what-is-serial-port/rs232-pinout/>.