

1. Comunicação de Dados

A distância que um dado sinal percorre em um computador varia de alguns milímetros, como no caso de conexões de um simples CI, até vários centímetros quando a conexão de sinais envolve, por exemplo, uma placa mãe com conectores para diversos circuitos. Para estas distâncias, o dado digital pode ser transmitido diretamente. Exceto em computadores muito rápidos, os projetistas não se preocupam com o formato e espessura dos condutores, ou com as características analógicas dos sinais de transmissão.

Freqüentemente, no entanto, os dados devem ser enviados para fora dos circuitos que constituem o computador. Nesses casos, as distâncias envolvidas podem ser enormes. Infelizmente, com o aumento das distâncias entre a fonte e o destino aumenta também a dificuldade de estabelecer uma transmissão de dados precisa. Isso é resultado de distorções elétricas dos sinais que trafegam através de condutores longos, e de ruídos adicionados ao sinal que se propagam através do meio de transmissão. Embora alguns cuidados devam ser tomados na troca de dados dentro de um computador, o grande problema ocorre quando dados são transferidos para dispositivos fora dos circuitos do computador. Nesse caso a distorção e o ruído podem tornar-se tão severos que a informação é perdida. A Comunicação de Dados estuda os meios de transmissão de mensagens digitais para dispositivos externos ao circuito originador da mensagem. Dispositivos Externos são geralmente circuitos com fonte de alimentação independente dos circuitos relativos a um computador ou outra fonte de mensagens digitais. Como regra, a taxa de transmissão máxima permissível de uma mensagem é diretamente proporcional a potência do sinal, e inversamente proporcional ao ruído. A função de qualquer sistema de comunicação é fornecer a maior taxa de transmissão possível, com a menor potência e com o menor ruído possível.

1.1 Canais de Comunicação

Um canal de comunicação é um caminho sobre o qual a informação pode trafegar. Ela pode ser definida por uma linha física (fio) que conecta dispositivos de comunicação, ou por um rádio, laser, ou outra fonte de energia radiante. Em comunicação digital, a informação é representada por bits de dados individuais, que podem ser encapsulados em mensagens de vários bits. Um byte (conjunto de 8 bits) é um exemplo de uma unidade de mensagem que pode trafegar através de um canal digital de comunicações. Uma coleção de bytes pode ser agrupada em um “frame” ou outra unidade de mensagem de maior nível. Esses múltiplos níveis de encapsulamento facilitam o reconhecimento de mensagens e interconexões de dados complexos.

Um canal no qual a direção de transmissão é inalterada é referida como **canal simplex**. Por exemplo, uma estação de rádio é um canal simplex porque ela sempre transmite o sinal para os ouvintes e nunca é permitido a transmissão inversa.

Um **canal half-duplex** é um canal físico simples no qual a direção pode ser revertida. As mensagens podem fluir nas duas direções, mas nunca ao mesmo tempo. Em uma chamada telefônica, uma parte fala enquanto a outra escuta. Depois de uma pausa, a outra parte fala e a primeira escuta. Falar simultaneamente resulta em sons que não podem ser compreendidos.

Um **canal full-duplex** permite que mensagens sejam trocadas simultaneamente em ambas as direções. Ele pode ser visto como dois canais simplex, um canal direto e um canal reverso, conectados nos mesmos pontos.

1.2 Comunicação Serial

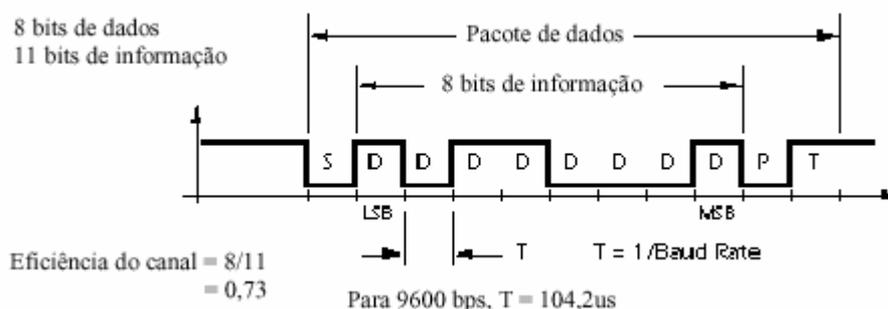
A maioria das mensagens digitais são mais longas que alguns poucos bits. Por não ser prático nem econômico transferir todos os bits de uma mensagem simultaneamente, a mensagem é quebrada em partes menores e transmitida seqüencialmente. A transmissão bit-serial converte a mensagem em um bit por vez através de um canal. Cada bit representa uma parte da mensagem. Os bits individuais são então rearranjados no destino para compor a mensagem original. Em geral, um canal irá passar apenas um bit por vez. A transmissão bit-serial é normalmente chamada de transmissão serial, e é o método de comunicação escolhido por diversos periféricos de computadores.

A transmissão byte-serial converte 8 bits por vez através de 8 canais paralelos. Embora a taxa de transferência seja 8 vezes mais rápida que na transmissão bit-serial, são necessários 8 canais, e o custo poderá ser maior do que 8 vezes para transmitir a mensagem. Quando as distâncias são curtas, é factível e econômico usar canais paralelos como justificativa para as altas taxas de transmissão. A interface Centronics de impressoras é um caso típico de transmissão byte-serial.

1.3 Taxa de Transferência (Baud Rate)

A taxa de transferência refere-se a velocidade com que os dados são enviados através de um canal e é medido em transições elétricas por segundo. Na norma EIA232, ocorre uma transição de sinal por bit, e a taxa de transferência e a taxa de bit (bit rate) são idênticas. Nesse caso, uma taxa de 9600 bauds corresponde a uma transferência de 9600 dados por segundo, ou um período de aproximadamente, 104 us (1/9600 s).

Outro conceito é a eficiência do canal de comunicação que é definido como o número de bits de informação utilizável (dados) enviados através do canal por segundo. Ele não inclui bits de sincronismo, formatação, e detecção de erro que podem ser adicionados a informação antes da mensagem ser transmitida, e sempre será no máximo igual a um.



Nota:

Baud: numero de mudanças do sinal por segundo.

Bits/ baud: numero de bits por baud.

Bits/s: numero de bits por segundo. Baud*bits por mudança de sinal

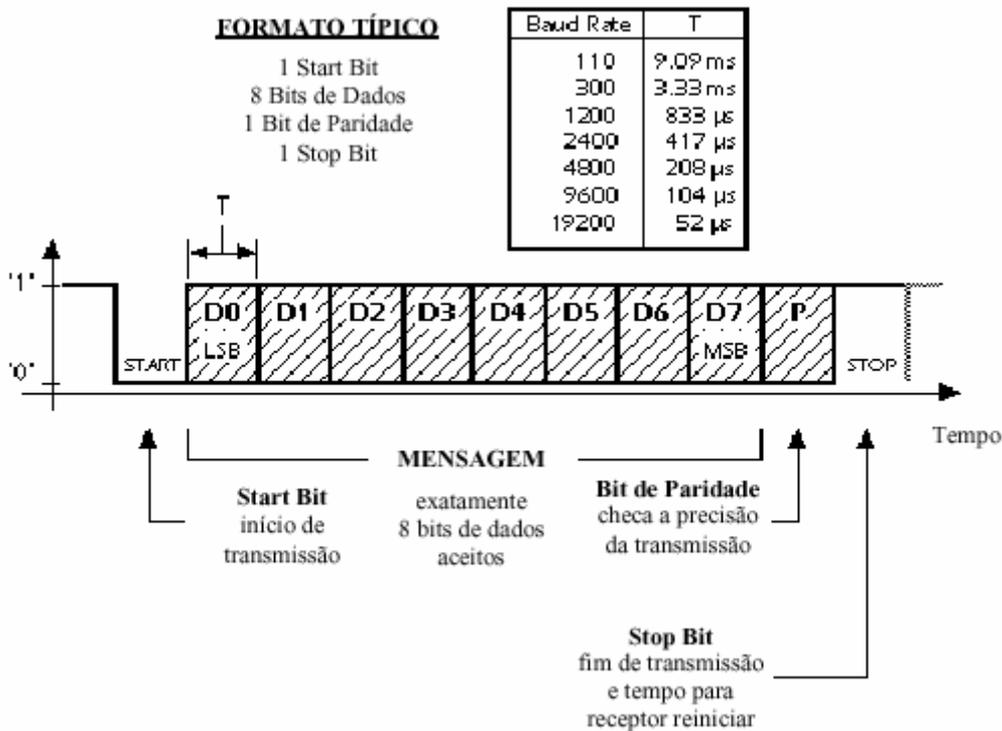
Bandwidth: É a largura da faixa. É a diferença entre a maior frequência e a menor frequência que o canal suporta. A velocidade de transmissão depende do bandwidth. Quanto maior o bandwidth maior a capacidade de transmissão

1.4 Transmissão Assíncrona x Transmissão Síncrona

Geralmente, dados serializados não são enviados de maneira uniforme através de um canal. Ao invés disso, pacotes com informação regulares são enviados seguidos de uma pausa. Os pacotes de dados binários são enviados dessa maneira, possivelmente com comprimentos de pausa variável entre pacotes, até que a mensagem tenha sido totalmente transmitida. O circuito receptor dos dados deve saber o momento apropriado para ler os bits individuais desse canal, saber exatamente quando um pacote começa e quanto tempo decorre entre bits. Quando essa temporização for conhecida, o receptor é dito estar sincronizado com o transmissor, e a transferência de dados precisa torna-se possível. Falhas na manutenção do sincronismo durante a transmissão irão causar a corrupção ou perda de dados. Duas técnicas básicas são empregadas para garantir a sincronização correta. Em sistemas síncronos, canais separados são usados para transmitir dados e informação de tempo. O canal de temporização transmite pulsos de clock para o receptor. Através da recepção de um pulso de clock, o receptor lê o canal de dado e armazena o valor do bit encontrado naquele momento. O canal de dados não é lido novamente até que o próximo pulso de clock chegue. Como o transmissor é responsável pelos pulsos de dados e de temporização, o receptor irá ler o canal de dados apenas quando comandado pelo transmissor, e portanto a sincronização é garantida.

Existem técnicas que compõem o sinal de clock e de dados em um único canal. Isso é usual quando transmissões síncronas são enviadas através de um modem. Dois métodos no qual os sinais de dados contém informação de tempo são: codificação NRZ (Non-Return-to-Zero) e a codificação Manchester.

Em sistemas assíncronos, a informação trafega por um canal único. O transmissor e o receptor devem ser configurados antecipadamente para que a comunicação se estabeleça a contento. Um oscilador preciso no receptor irá gerar um sinal de clock interno que é igual (ou muito próximo) ao do transmissor. Para o protocolo serial mais comum, os dados são enviados em pequenos pacotes de 10 ou 11 bits, dos quais 8 constituem a mensagem. Quando o canal está em repouso, o sinal correspondente no canal tem um nível lógico '1'. Um pacote de dados sempre começa com um nível lógico '0' (start bit) para sinalizar ao receptor que uma transmissão foi iniciada. O "start bit" inicializa um temporizador interno no receptor avisando que a transmissão começou e que serão necessários pulsos de clocks. Seguido do start bit, 8 bits de dados de mensagem são enviados na taxa de transmissão especificada. O pacote é concluído com os bits de paridade e de parada ("stop bit").



1.5 Conjunto de Caracteres ASCII

Os caracteres enviados através de uma interface serial geralmente seguem o padrão ASCII (American Standard Code for Information Interchange) de 7 bits.

1.6 Checksum e Paridade

Ruídos e distúrbios elétricos momentâneos podem causar mudanças nos dados quando estão trafegando pelos canais de comunicação. Se o receptor falhar ao detectar isso, a mensagem recebida será incorreta, resultando em conseqüências possivelmente sérias. Como uma primeira linha de defesa contra erros de dados, eles devem ser detectados. Se um erro pode ser sinalizado, pode ser possível pedir que o pacote com erro seja reenviado, ou no mínimo prevenir que os dados sejam tomados como corretos. Se uma redundância na informação for enviada, 1 ou 2 bits de erros podem ser corrigidos pelo hardware no receptor antes que o dado chegue ao seu destino.

O bit de paridade é adicionado ao pacote de dados com o propósito de detecção de erro. Na convenção de paridade-par (“even-parity”), o valor do bit de paridade é escolhido de tal forma que o número total de dígitos ‘1’ dos dados adicionado ao bit de paridade do pacote seja sempre um número par. Na recepção do pacote, a paridade do dado precisa ser recomputada pelo hardware local e comparada com o bit de paridade recebido com os dados. Se qualquer bit mudar de estado, a paridade não irá coincidir, e um erro será detectado. Se um número par de bits for trocado, a paridade coincidirá e o dado com erro será validado. Contudo, uma análise estatística dos erros de comunicação de dados tem mostrado que um erro com bit simples é muito mais provável que erros em múltiplos bits na presença de ruído randômico. Portanto, a paridade é um método confiável de detecção de erro.

Dado	Bit de Paridade
1 0 1 1 0 0 1 0	0
1 0 0 0 1 0 1 0	1

Outro método de detecção de erro envolve o cálculo de um “checksum” quando mensagens com mais de um byte são transmitidas pelo canal de comunicação. Nesse caso, os pacotes que constituem uma mensagem são adicionados aritmeticamente. Um número de checksum é adicionado a seqüência do pacote de dados de tal forma que a soma dos dados mais o checksum é zero.

Quando recebido, os dados devem ser adicionados pelo processador local. Se a soma do pacote der resultado diferente de zero, ocorreu um erro. Na ocorrência de erros é improvável (mas não impossível) que qualquer corrupção de dados resultem em checksum igual a zero.

$$\begin{array}{r}
 10110001 \\
 10000110 \\
 + 01001100 \text{ Dados} \\
 11111111 \\
 10100000 \\
 \hline
 001100100010 \text{ Soma Aritmética} \\
 \\
 00100010 \text{ Soma truncada - 8 bits} \\
 + 11011110 \text{ Checksum (complemento de 2)} \\
 \hline
 00000000 \text{ Soma + Checksum = 0}
 \end{array}$$

Podem ocorrer erros que não sejam apenas detectados, mas também sejam corrigidos se código adicional for adicionado a seqüência de dados do pacote. A correção de erros em uma transmissão, contudo, abaixa a eficiência do canal, e o resultado é uma queda na transmissão.

2. Interface Serial RS232 (EIA232)

RS é uma abreviação de “Recommended Standard”. Ela relata uma padronização de uma interface comum para comunicação de dados entre equipamentos, criada no início dos anos 60, por um comitê conhecido atualmente como “Electronic Industries Association” (EIA). Naquele tempo, a comunicação de dados compreendia a troca de dados digitais entre um computador central (mainframe) e terminais de computador remotos, ou entre dois terminais sem o envolvimento do computador. Estes dispositivos poderiam ser conectados através de linha telefônica, e conseqüentemente necessitavam um modem em cada lado para fazer a decodificação dos sinais.

Dessas idéias nasceu o padrão RS232. Ele especifica as tensões, temporizações e funções dos sinais, um protocolo para troca de informações, e as conexões mecânicas. A mais de 30 anos desde que essa padronização foi desenvolvida, a EIA publicou três modificações. A mais recente, EIA232E, foi introduzida em 1991. Ao lado da mudança de nome de RS232 para EIA232, algumas linhas de sinais foram renomeadas e várias linhas novas foram definidas. Embora tenha sofrido poucas alterações, muitos fabricantes adotaram diversas soluções mais simplificadas que tornaram impossível a simplificação da padronização proposta. As maiores dificuldades encontradas pelos usuários na utilização da interface RS232 incluem pelo menos um dos seguintes fatores:

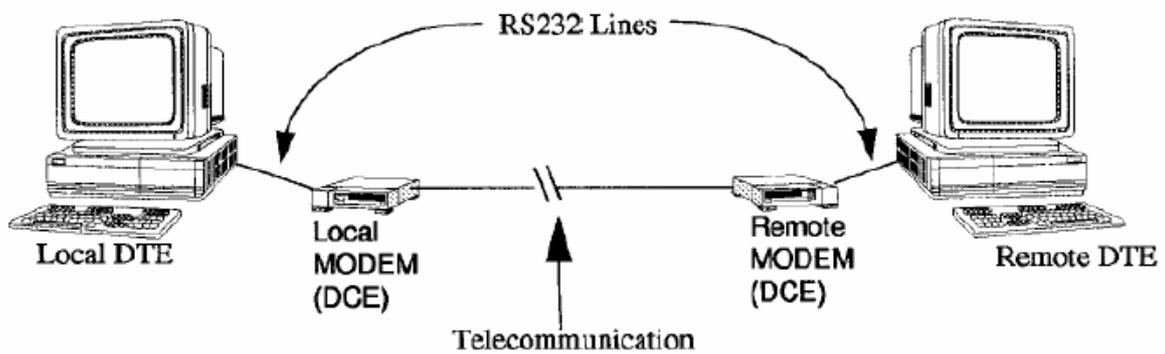
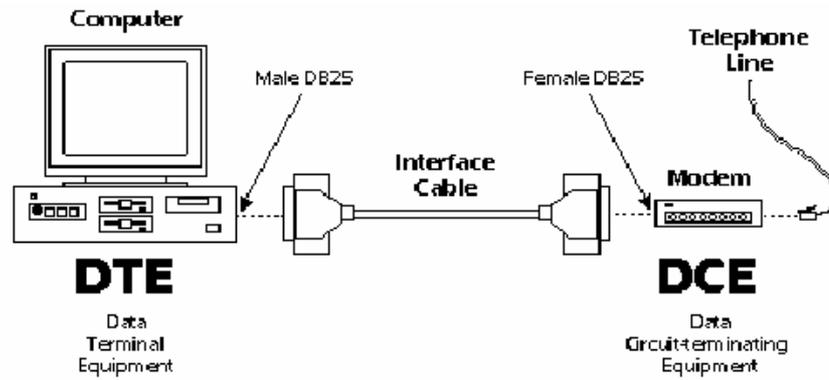
A ausência ou conexão errada de sinais de controle, resultam em estouro do buffer (“overflow”) ou travamento da comunicação.

Função incorreta de comunicação para o cabo em uso, resultam em inversão das linhas de Transmissão e Recepção, bem como a inversão de uma ou mais linhas de controle (“handshaking”).

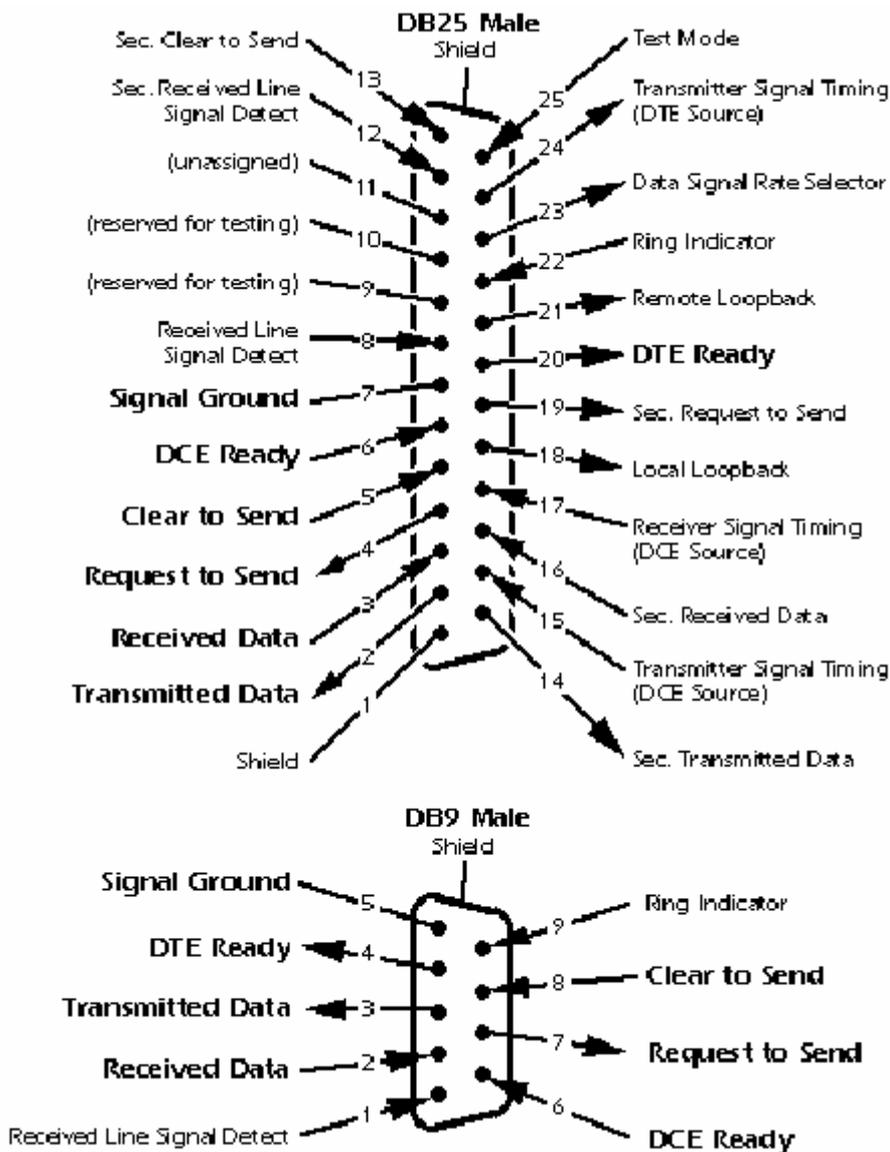
Felizmente, os drivers utilizados são bastante tolerantes aos abusos cometidos, e os CIs normalmente sobrevivem.

2.1 Definição de Sinais

Se a norma EIA232 completa for implementada, o equipamento que faz o processamento dos sinais é chamado DTE (Data Terminal Equipment – usualmente um computador ou terminal), tem um conector DB25 macho, e utiliza 22 dos 25 pinos disponíveis para sinais ou terra. O equipamento que faz a conexão (normalmente uma interface com a linha telefônica) é denominado de DCE (Data Circuit-terminating Equipment – usualmente um modem), tem um conector DB25 fêmea, e utiliza os mesmos 22 pinos disponíveis para sinais e terra. Um cabo de conexão entre dispositivos DTE e DCE contém ligações em paralelo, não necessitando mudanças na conexão de pinos. Se todos os dispositivos seguissem essa norma, todos os cabos seriam idênticos, e não haveria chances de haver conexões incorretas.



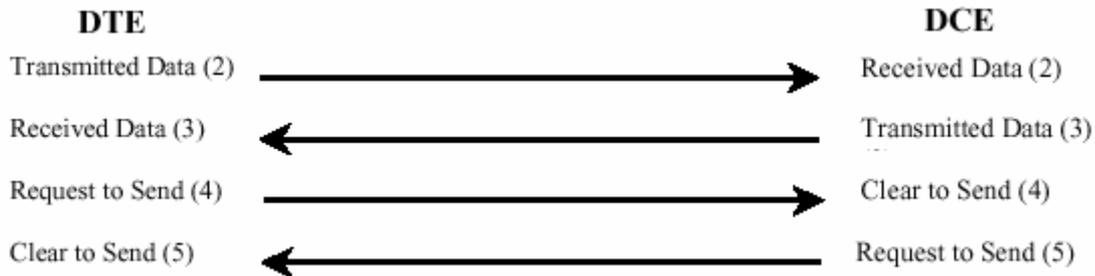
Na figura a seguir é apresentada a definição dos sinais para um dispositivo DTE (usualmente um micro PC). Os sinais mais comuns são apresentados em negrito.



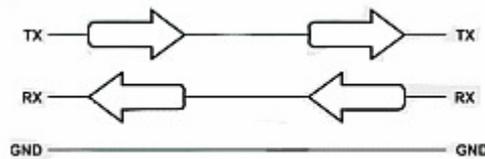
Diversos sinais são necessários para conexões onde o dispositivo DCE é um modem, e eles são utilizados apenas quando o protocolo de software os emprega. Para dispositivos DCE que não são modem, ou quando dois dispositivos DTE são conectados diretamente, poucos sinais são necessários. Deve-se notar que nas figuras apresentadas existe um segundo canal que inclui um conjunto de sinais de controle duplicados. Este canal secundário fornece sinais de gerenciamento do modem remoto, habilitando a mudança de taxa de transmissão durante a comunicação, efetuando um pedido de retransmissão se erros de paridade forem detectados, e outras funções de controle.

Os sinais de temporização de transmissão e recepção são utilizados somente quando o protocolo de transmissão utilizado for síncrono. Para protocolos assíncronos, padrão 8 bits, os sinais de temporização externos são desnecessários. Os nomes dos sinais que implicam em um direção. Como "Transmit Data" e "Receive Data", são nomeados do ponto de vista dos dispositivos DTE. Se a norma EIA232 for seguida a risca, estes sinais terão o mesmo nome e o mesmo número de pino do lado do DCE. Infelizmente, isto não é feito na

prática pela maioria dos engenheiros, provavelmente porque em alguns casos torna-se difícil definir quem é o DTE e quem é o DCE. A figura a seguir apresenta a convenção utilizada para os sinais mais comuns.



2.2 Padrão Serial RS232C



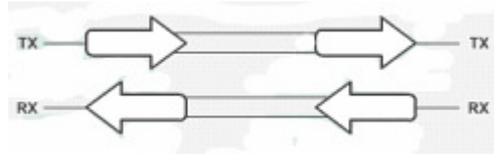
Sinais: A RS232C possui basicamente dois sinais de comunicação sendo o Tx aquele que envia e o Rx o que recebe. O nível do diferencial binário é comparado com a tensão do terceiro sinal GND. Há outros sinais que podem ser utilizados para controle do fluxo e dos pontos da comunicação. Nível alto entre -3 e -25V, nível baixo entre +3 e +25V.

Número max de equipamentos: 2 em uma conexão ponto a ponto.

Distância Max.: Até 20 metros para pontos energizados

3. Padrão Serial RS422 e RS485 RS-422/423

Define uma interface balanceada, mas não define um conector físico. Fabricantes que aderiram a este padrão usam muitos conectores diferentes, incluindo os terminais de parafusos, DB9, DB25 com pinagem não padronizada, DB25 com padrão RS-530 e DB37 com padrão RS-449. O RS-422 é comumente usado em comunicações ponto a ponto realizadas por um driver dual-state. As transmissões podem ir a grandes distâncias e altas velocidades.

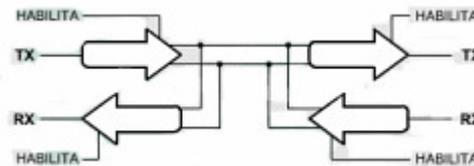


Sinais: A RS422 possui sinais de comunicação Tx+, Rx+, Tx- e RX-, sendo o Tx aquele que envia e Rx o que recebe. O modo de transmissão é por diferencial elétrico. Pode utilizar outros sinais para controle.

Número max de equipamentos: 10 em uma conexão de barramento único.

Distância Max.: Até 1200 metros para o último ponto.

RS-485 - é semelhante ao RS-422, exceto pelo fato dos drivers associados serem tri-state e não dual-state. Pode ser utilizado em aplicações multiponto em que um computador controla muitos dispositivos diferentes. Até 32 dispositivos podem ser conectados com o RS-485.



Sinais: A RS485 possui sinais de comunicação Tx+/Rx-, Tx-/ RX+, sendo o Tx aquele que envia e o Rx o que recebe. O modo de transmissão é por diferencial elétrico. Pode utilizar outros sinais para controle.

Número max de equipamentos: 32 em uma conexão de barramento único.

Distância Max.: Até 1200 metros para o último ponto.

4. USB – Barramento Serial Universal

4.1 Estrutura de Software

O barramento serial universal, especificado pelas empresas líderes no mercado de computadores pessoais (Compaq, DEC, IBM, Intel, Microsoft, NEC e outras), permite uma expansão externa do PC praticamente ilimitada. Com o USB, os usuários aproveitam os benefícios da arquitetura plug-and-play, ou seja, não necessitam mais de efetuar configurações de recursos de hardware, como nos quebra-cabeças dos "dip-switches" e "jumpers", para a definição de IRQ's, canais de DMA ou endereços de I/O.

O USB utiliza um conector universal que permite ao usuário instalar e remover periféricos sem sequer abrir o computador. E ainda, com a característica de inserção e

remoção automática, os periféricos podem ser instalados e removidos a qualquer momento, mesmo com o computador ligado e inicializado. Além da facilidade de utilização de periféricos convencionais, o USB abre caminho para novas aplicações, como a integração PC/telefonia e jogos multiusuários.

Dois importantes atributos do USB são também destacados: a compatibilidade universal, pois nada impede que o USB seja aproveitado por outra arquitetura, e a simplicidade no projeto de periféricos, pois são eliminados diversos custos, como o de interfaces auxiliares (ex: alguns scanners e CD ROM precisam de uma interface SCSI).

O USB pode ser usado com a maioria dos periféricos de PC's, tais como: controladoras de vídeo, drives de CD ROM, joysticks, unidades de fita, drives de disco flexível externos, scanners ou impressoras. A taxa de transmissão especificada de 12 Megabits/s também acomoda uma nova geração de periféricos, incluindo os produtos baseados em vídeo (ex: câmeras digitais).

4.2 Uma Visão Geral do Funcionamento

Quanto à organização das camadas de software necessárias para operar-se um dispositivo, o sistema USB HOST é composto por vários níveis de hardware e softwares, conforme mostrado na Figura 1.

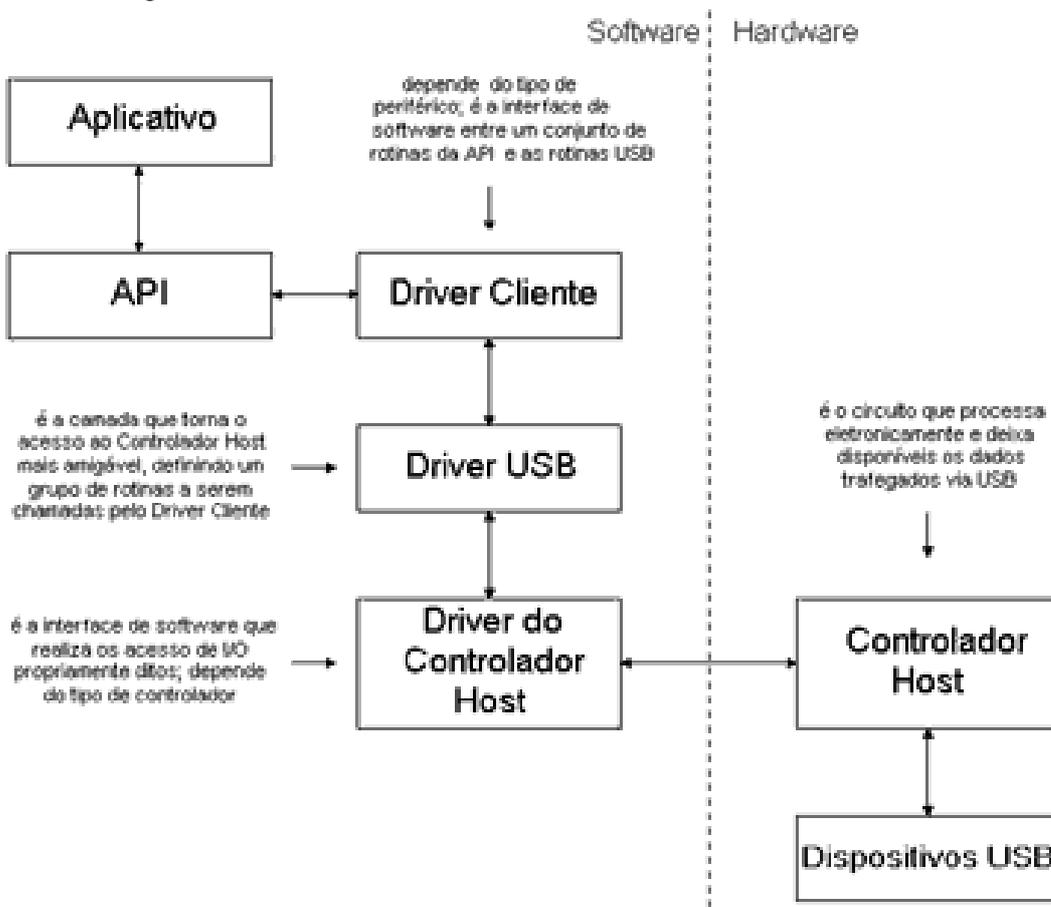


Figura 1: Camadas de software necessárias para operarem-se dados via USB.

Conforme a Figura 1 indica, um aplicativo requer o acesso a um periférico USB da forma padrão, como é feita para os periféricos comuns: chama funções da API.

Num segundo estágio, a API chama rotinas do driver Cliente do periférico USB instalado. Este driver traduz os comandos da API para comandos USB. O driver Cliente é geralmente parte do sistema operacional ou vem instalável com o dispositivo USB.

A terceira camada de software indicada é o driver USB (USB D), que é aquela que dá ao sistema operacional o suporte ao USB.

A quarta camada de interesse é o driver do controlador HOST (HCD), que funciona a nível de Kernel do sistema operacional. O HCD provê o nível de software entre o hardware do controlador HOST e o USB D. É esta camada que realiza os acessos de I/O necessários para a operação do dispositivo USB. O HCD interpreta as chamadas do USB D e constrói uma lista de estruturas, um descritor de transferências, uma fila principal e um buffer de dados para o controlador HOST.

A Figura 1 apresenta também duas camadas de hardware. A primeira delas é o Controlador HOST (HC), que é o circuito onde serão feitas as conexões de todos os dispositivos USB. Tal circuito executa eletronicamente os comandos programadas pelo HCD, além de registrar o estado das transações do USB. Sua operação é gerenciada pelo HCD.

A segunda delas é constituída pelo conjunto de Dispositivos USB conectados, que são os periféricos que usam esse tipo de barramento.

Uma característica extremamente inovadora do USB é a possibilidade de conectar-se um novo dispositivo durante a execução do aplicativo. Neste caso, o controlador HOST detecta a conexão e envia uma mensagem ao HCD para avisá-lo do fato. Em seguida, o HCD faz a mesma notificação ao driver USB (USB D). Este, então, inicializa o driver cliente do periférico conectado e, em seguida, torna-o operacional, de forma que o aplicativo já possa dispor de seus recursos. Tal seqüência está ilustrada na Figura 2.

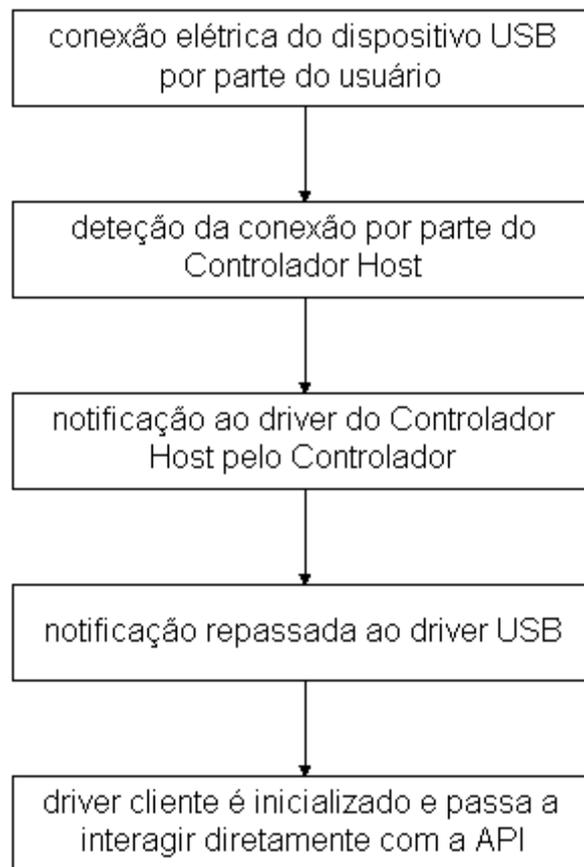


Figura 2: Seqüência de eventos desde a conexão da placa até a inicialização dos drivers.

- Para verificar se o periférico comprado possui interface USB, verificar se o cabo de conexão tem forma semelhante à da Figura 3.



Figura 3: Plug USB.

A tecnologia USB reúne qualidades que tornam o PC mais flexível e capaz de explorar ainda mais suas potencialidades em termos de aplicação prática . O que antes limitava o PC a uns poucos periféricos, hoje o lança num ambiente de grande multiplicidade sem grandes complicações tecnológicas, facilitando a vida dos usuários.

4.3 Estrutura Elétrica do USB

O Universal Serial Bus (USB) é uma nova filosofia de barramento serial para o fluxo de dados entre um computador e dispositivos periféricos. O USB foi projetado para preencher certas lacunas deixadas, até então, pelos outros barramentos seriais. Ele oferece:

Melhor integração entre a computação e os sistemas de comunicações, focalizando-se na CTI (Computer Telephony Integration);

Maior comodidade para o usuário, que não necessita de configurar o dispositivo, pois o próprio sistema se encarrega desta tarefa; o USB incorpora a filosofia plug-and-play;

Expansão no número de portas; ele pode endereçar até 127 dispositivos.

O sistema USB é composto pelo Controlador Host USB, por dispositivos USB e por interconexões. A Figura 1 mostra a topologia de um sistema USB. O host é responsável por: detectar a inserção e a remoção de um dispositivo, por gerenciar o fluxo de dados e de controle, monitorando o estado das transferências, e por controlar a interface elétrica entre ele e os dispositivos.

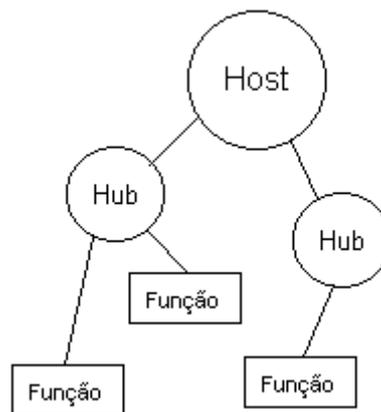


Figura 1: Topologia de um sistema USB

Um dispositivo USB pode ser de dois tipos: função ou hub. A função é capaz de transmitir ou receber dados ou informações de controle pelo barramento. Ela serve para aumentar a capacidade do sistema. Exemplos de funções são: mouse, teclado, impressora e adaptador telefônico como um ISDN. Cada função contém informações descrevendo suas capacidades e os recursos dos quais necessita.

O hub é o elemento chave na topologia USB, pois é ele que permite a expansão do número de conexões do sistema. Cada hub converte um ponto de conexão em outros múltiplos pontos. A arquitetura USB permite o uso de múltiplos hubs.

O USB permite dois modos de comunicação: um de alta velocidade, operando a 12 Mb/s, e outro de baixa velocidade, a 1,5 Mb/s. O modo de baixa velocidade visa a atender a um pequeno número de dispositivos com largura de banda estreita, como mouses. Cada função é responsável por indicar em que modo irá operar.

4.4 O barramento físico

O barramento físico é composto de um cabo com quatro fios: VBus, D+, D- e GND. O fio VBus é o meio de fornecimento de alimentação para os dispositivos que necessitarem dela. Em um sistema USB, existem hubs e funções que possuem alimentação própria e hubs e funções que são alimentados pelo barramento através de VBus. VBus é nominalmente +5 V. Para aplicações de alta velocidade, os fios D+ e D- são entrelaçados. Os dados são transmitidos através de D+ e D- por meio de diferenças de tensão entre eles. O USB usa uma codificação NRZI.

Os cabos são conectados aos dispositivos conforme ilustrado na Figura 2. A posição dos resistores de pull-up muda dependendo de tratar-se de alta ou baixa velocidade. Quando não existe função conectada ao hub, os resistores de pull-down fazem com que ambos D+ e D- fiquem abaixo de um valor de tensão de limiar para a detecção da presença do dispositivo. Se essa condição persistir por mais que 2,5 microssegundos, é caracterizada a desconexão do dispositivo. A conexão de um dispositivo é caracterizada pela situação oposta, ou seja, quando apenas uma das linhas é levada além da tensão de limiar e esta situação persiste por mais de 2,5 microssegundos.

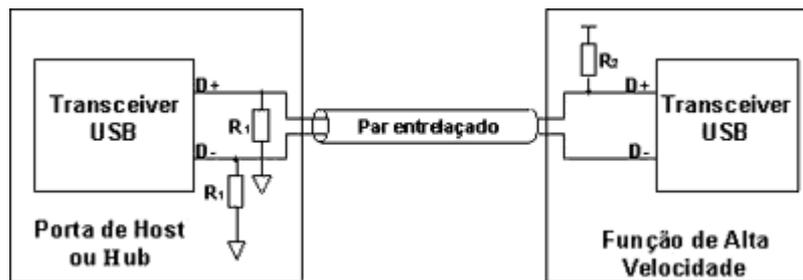


Figura 2: Esquema Físico de Conexão do Barramento

5. O Controlador Programável em Rede

5.1 Introdução

Da mesma forma que o processamento de dados e as comunicações estão sendo usadas por um número cada vez maior de empresas para automatizar seus procedimentos administrativos e as tarefas de escritório, estão também sendo usados nas usinas e industriais para controlar seus processos de produção e suas linhas de montagem. Três classes de aplicações podem ser visualizadas nesse ambiente:

a) Aplicações típicas de processamento de dados como planejamento da produção, controle de estoque, faturamento, etc.

b) Aplicações específicas de controle do processo industrial em siderúrgicas, fábricas de papel, de produtos químicos, de cimento, distribuição de energia elétrica, etc.

c) Aplicações em indústrias com uma linha de produção que deve ser adaptada dinamicamente para produzir diversos tipos de produtos, atender pedidos urgentes de mudança nas quantidades em produção, etc. Os dois últimos tipos de aplicação diferem do primeiro basicamente por serem aplicações em tempo real, ou seja, devem responder a eventos que exigem resposta em um prazo limitado, que se for ultrapassado poderá constituir uma falha no sistema.

O controle de um processo de produção envolve a necessidade de medida através de sensores, de certas variáveis do processo (temperatura, pressão, vazão, etc.), seu processamento e controle através dos atuadores, que alteram o estado do processo abrindo uma válvula, partindo um motor, ligando um aquecedor, etc. Um sistema de controle em tempo real tem como responsabilidade o processo de produção de grandes fábricas que representam vultosos investimentos. Eles devem, portanto, ter as seguintes características:

a) Isenção de erros - Um sistema em tempo real deve produzir sempre resultados corretos porque, ao contrário de sistemas convencionais, muitas vezes a recuperação manual de uma situação de erro não é possível.

b) Presteza - Num sistema convencional, é melhor produzir uma resposta mesmo que o limite desejado de tempo seja excedido. Já num sistema em tempo real, uma resposta dada além do tempo esperado é inútil.

c) Robustez - Os componentes do sistema devem ter um tempo médio entre falhas muito grande. Para isso devem ser projetados utilizando técnicas de tolerância a falhas. Caso ocorram falhas, o sistema deve ter a possibilidade de continuar funcionando, mesmo de forma limitada.

d) Flexibilidade - Qualquer sistema de maior vulto é afetado ao longo do tempo pelo surgimento de novas tecnologias, pela mudança nas especificações ou pela necessidade de melhorar seu desempenho em relação a especificações existentes. Um sistema flexível permite a identificação dos módulos afetados por estas mudanças e sua eventual adaptação ou substituição, sem impactar o sistema como um todo. Esses requisitos são mais

facilmente atendidos por uma arquitetura distribuída. Os sistemas de automação distribuídos são formados por equipamentos baseados em microprocessadores que controlam um certo número de sensores e atuadores. Uma rede de comunicação conecta estes módulos entre si e aos módulos de operação e supervisão. O Controlador Programável (CLP) é o equipamento básico no controle de processos industriais e automação de máquinas.

Disponível em um número ilimitado de configurações e modelos, o CLP trata diretamente com as informações da planta - Leitura de sensores analógicos e digitais, atuação analógica ou digital, processamento lógico-aritmético em alta velocidade, etc. Associado a estas características, os Controladores Programáveis possuem recursos de comunicação que permitem a montagem de sistemas distribuídos, no porte adequado a cada aplicação. Outros módulos de controle importantes são os sensores e atuadores inteligentes.

Um produto deste tipo tem uma tarefa específica (medição de temperatura, de fluxo, de pressão, etc.), possuindo a capacidade de integração direta à rede através de um canal de comunicação. Um módulo de operação típico é um interface Homem-Máquina. Baseado nas informações apresentadas em um display, o operador tem acesso ao estado do processo, pode alterar pontos de verificação das variáveis do processo, parâmetros dos algoritmos de controle e aquisição de dados, além de outras funções como visualização da tendência das variáveis e o exame em detalhe de algum ponto do processo. O módulo de supervisão é responsável por funções mais complexas como a otimização do desempenho global, sequenciamento e escalonamento de tarefas, recuperação automática de falhas, etc. A aquisição de dados é, pela própria natureza da aplicação, distribuída pela fábrica ou usina.

A utilização de módulos com capacidade de processamento local, controlando os sensores, permite que os dados sejam amostrados e comparados com os valores desejados localmente. Apenas médias ou valores discrepantes precisam ser enviados aos módulos de operação e supervisão, ao contrário do que acontece nos sistemas centralizados onde toda medida precisa ser enviada à central para processamento. Com isso o tráfego é menor, diminuindo a demora na resposta em situações de muita carga e aumentando a confiabilidade geral do sistema.

Resumidamente, as redes de equipamentos de automação podem ser consideradas como soluções para o problema de interconexão e compartilhamento de recursos computacionais (hardware e software) geograficamente dispersos.

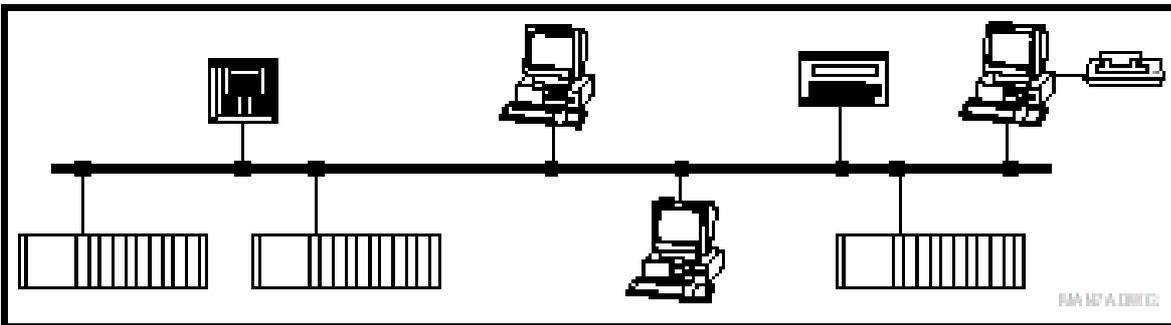
6 - Arquiteturas básicas - definições (LAN X WAN)

Definindo desta forma uma rede como uma estrutura de produtos (hardware e software), interligados de acordo com um padrão pré-estabelecido para satisfazer os requisitos dos sistemas distribuídos, cabe aqui distinguir duas arquiteturas básicas:

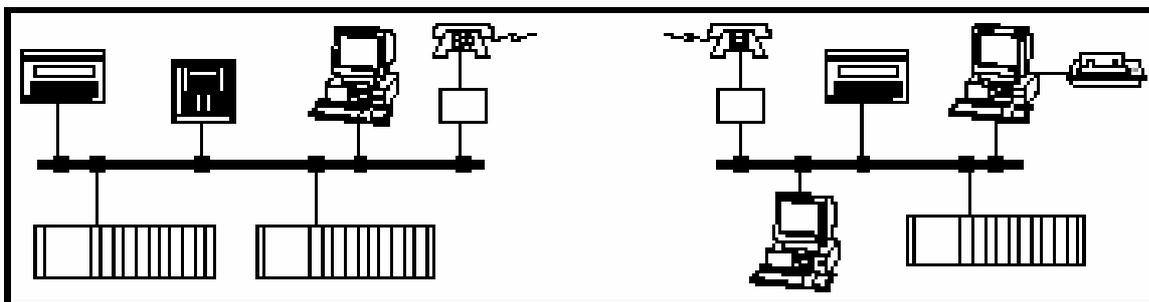
A Rede Local (Local Area Network) - A característica básica da rede local é a que determina que os equipamentos interligados estão confinados a uma área geograficamente limitada. Normalmente o sistema completo pertence a uma única organização e seu raio de ação está limitado a, no máximo, alguns quilômetros. Resumidamente as características gerais desta estrutura são:

- Dimensões moderadas
 - Alta capacidade de transmissão de informação
- Alta confiabilidade na comunicação

- Conectividade total ou quase total entre as estações



A Rede de Longa Distância (Wide Area Network) - As redes de longa distância estendem as características das redes locais no que se refere principalmente à área de abrangência. Através de recursos de telecomunicações, uma rede desse tipo pode ter dimensões globais, com um número indeterminado e muito grande de estações interligadas.



Os próximos capítulos (3 a 8) tratarão especificamente da concepção e projeto de redes locais utilizando equipamentos comerciais. O capítulo 9 apresenta uma visão geral das Redes de Longa Distância, como uma extensão do conceito de Rede Local.

7 - Funcionamento básico da rede

O processo de comunicação entre as diversas estações de uma rede pressupõe a existência de um conjunto de regras e convenções que permita disciplinar a troca de informações. Essas regras comuns são chamadas Protocolos de Comunicação. O conjunto destas regras compõem o que se chama Arquitetura da Rede. Para o projeto prático de uma rede, é importante compreender os aspectos básicos da arquitetura, descritos a seguir:

7.1 - Topologia

A topologia de uma rede corresponde à estrutura de interconexão física das várias estações que a compõem. Cada ponto de conexão de uma estação à rede é denominado um Nó de Comunicação.

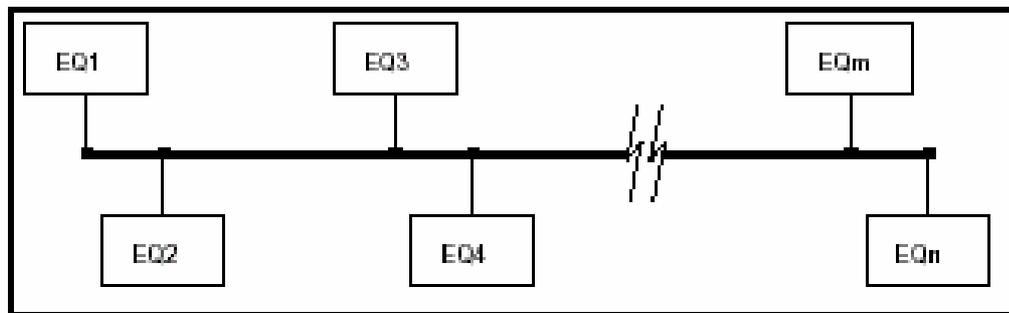
Estudaremos agora a Topologia denominada Rede em Barramento (Bus Network). Nesta topologia, o meio físico de transmissão é composto por um único segmento, compartilhado por todas as estações conectadas. Esta solução simplifica consideravelmente o problema de instalação e manutenção do meio físico:

Permite quantificar e limitar mais facilmente os tempos de espera para acesso a rede.

Novas estações podem ser adicionadas ou removidas da rede com facilidade, o que se constitui numa importante vantagem em sistemas industriais. Por outro lado, esta mesma característica tende a limitar o alcance físico e o número de estações permitidas na rede.

Um ponto fundamental nesta estrutura é a necessidade de um mecanismo de software que discipline o acesso das diversas estações ao meio físico compartilhado.

Na topologia em barramento, dois fios interconectam todas as estações, permitindo que qualquer equipamento se comunique com todos os outros.

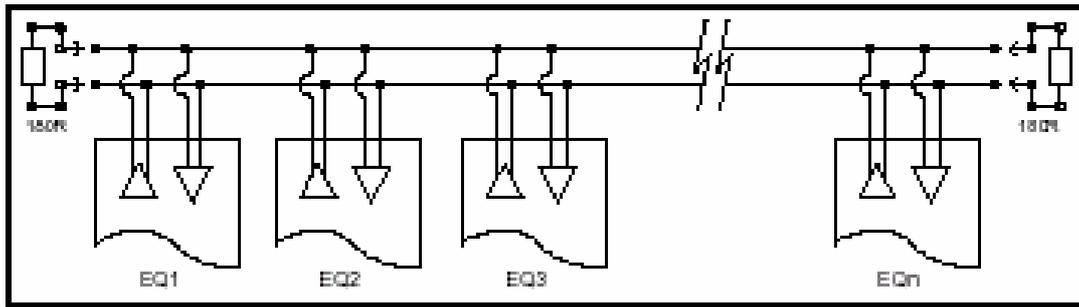


7.2 - Meio de transmissão e características elétricas

O Meio de Transmissão corresponde ao suporte físico da comunicação, estruturado de acordo com a topologia adotada. Em geral, é previsto a utilização de cabos de cobre na forma de par trançado. Normalmente é usado um cabo blindado com um ou dois pares trançados. Os componentes de interface de comunicação são projetados para se ajustar as características elétricas deste tipo de cabo, previstas na norma internacional EIA-RS485. Um cabo do tipo par trançado blindado é suficiente para interligar as estações. Os drivers codificam o sinal na forma de tensão diferencial.

Características elétricas:

- Ligação dos equipamentos através de par trançado
- Transmissão através de sinal diferencial de tensão (+-5V)
- Impedância da linha: 180 Ohms. Em linhas longas (>50m), são utilizados dois resistores de terminação deste valor nas pontas.
- Transmissores/Receptores ligados em paralelo
- Alcance de 1200 metros a no máximo 100 kbauds
- Capacidade de conexão de até N equipamentos



8 - O protocolo de comunicação – Definição Geral

Esta norma pretende padronizar a forma de comunicação entre os diferentes produtos usados para automação e controle de processos. O objetivo é possibilitar e uniformizar as regras para comunicação via canal serial entre diferentes produtos, desenvolvidos para aplicações as mais variadas.

Exemplos de produtos:

- Controladores programáveis de pequeno porte
- Controladores programáveis de médio/grande porte
- Unidades de interface Homem-Máquina
- Computadores para controle de processo IBM-PC ou similares
- Módulos dedicados para comunicação, embutidos no computador
- Displays e I/O's remotos
- Controladores de processo remotos

A estrutura básica de aplicação do protocolo é aquela apresentada na figura do item 7.2. As características fundamentais da associação entre o protocolo e esta ligação física são as seguintes:

- a) Todos os equipamentos são ligados da mesma forma ao barramento.
- b) Cada equipamento (nó) recebe um endereço lógico para referência, que independe da sua localização física.
- c) Não existe unidade mestre no sistema.
- d) Todos os equipamentos podem ter acesso a qualquer outro.

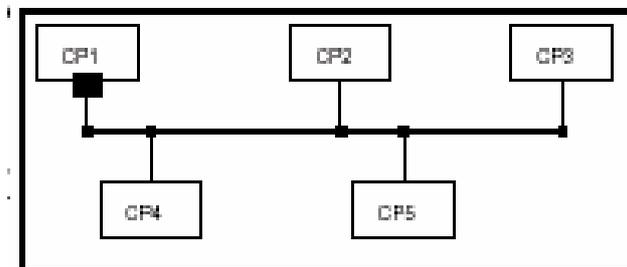
9. O MÉTODO DE PASSAGEM DE FICHA (TOKEN PASSING)

Ao ser inicializado o sistema, nenhum nó possui a ficha. A ficha é o elemento lógico que habilita um equipamento (nó) a solicitar um serviço de comunicação. Todos os nós tem a visão do que está acontecendo no barramento (linha). Como não há atividade na Linha, os equipamentos entram no modo "Criação de Ficha" (CF). No modo CF, cada nó aguarda um determinado tempo, após o qual, se não houver atividade na linha, este cria uma ficha. Por definição, o tempo de espera depende do endereço lógico do nó, de acordo com a tabela:

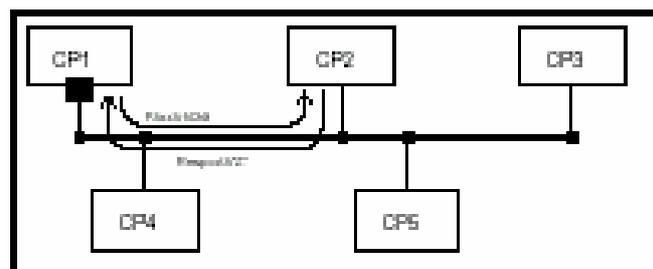
Código do nó	Tempo em ms
1	1000
2	1050
3	1100
4	1150
"	"
"	"
30	2450

A seguir será apresentado o mecanismo de circulação das mensagens e arbitragem das tarefas de comunicação através de um exemplo:

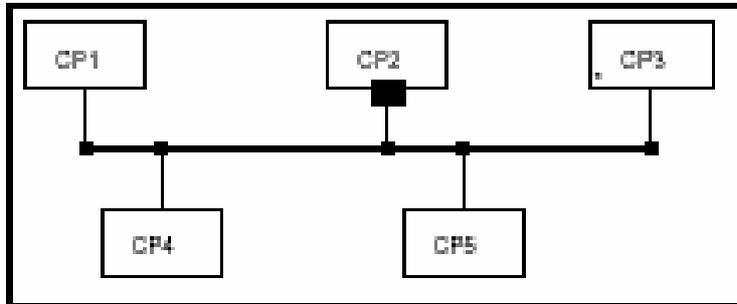
a) O diagrama da figura abaixo mostra o sistema 1,0 seg. após a inicialização:
 A ficha acabou de ser criada, habilitando o CLP1 a solicitar um serviço de comunicação.



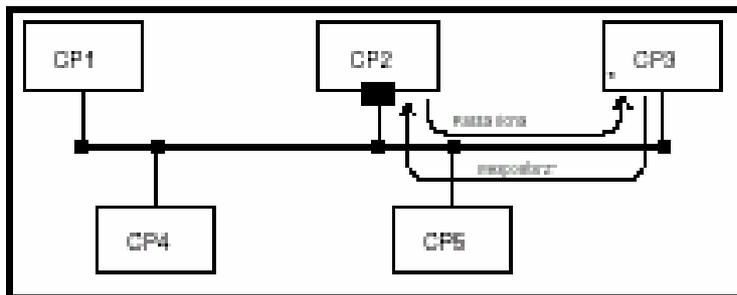
b) Vamos considerar que o CLP1 nada tem a transmitir. Nesta caso, ele deve passar a ficha imediatamente para o nó seguinte da rede:



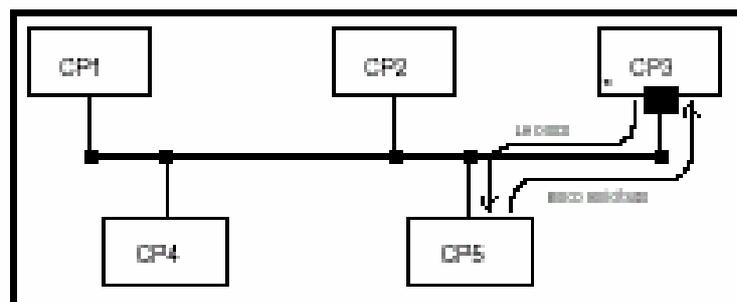
c) Após o CLP1 receber a confirmação do sucesso na passagem da ficha, esta passa para o CLP2. Nesse momento, o programa do usuário no CLP3 está solicitando a leitura de um bloco de memória no CLP5 (marcado com *). Essa requisição fica armazenada para execução posterior:



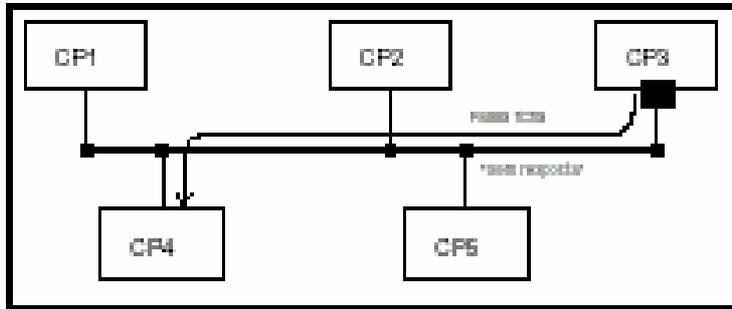
d) O CLP2 não tem nenhuma tarefa a executar. Em vista disso, ele executa imediatamente uma transferência de ficha para o CLP3:



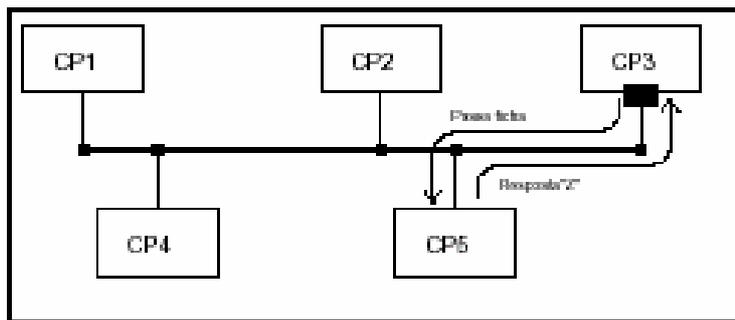
e) O CLP3 está com a solicitação pendente. Agora de posse da ficha, ele executa a operação de aquisição no CLP5. O CLP5 responde de acordo com a operação e parâmetros solicitados:



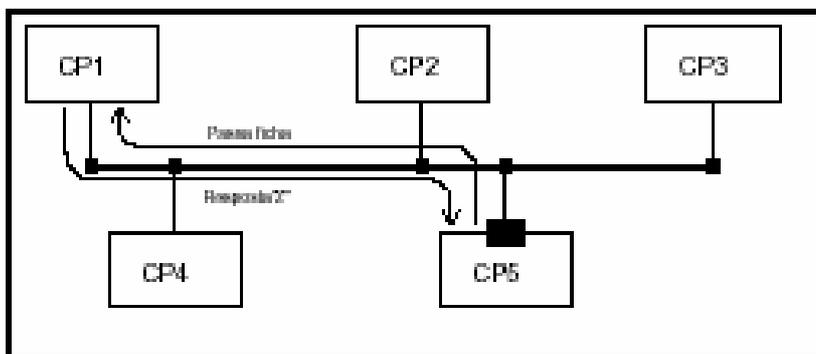
f) Imediatamente após o final da resposta, o CLP3 passa a ficha para o CLP4. Por qualquer motivo, o CLP4 está desligado e não irá recebê-la:



g) Não havendo resposta no CLP4 ante a tentativa de passagem da ficha, o CLP3 irá tentar passá-la adiante para o próximo na seqüência:



h) A ficha agora está com o CLP5. Como este equipamento não tem nenhuma operação por fazer e não há próximo nó na seqüência, a ficha é passada para o CLP1, fechando o ciclo:



Considerações especiais:

a) Caso haja um erro qualquer de paridade, framing ou overrun durante a execução de uma tarefa, a operação corrente é suspensa e a ficha passada ao equipamento seguinte. Os equipamentos devem manter um registro do número de eventos desse tipo em seu software operacional.

b) Os eventos previsíveis que causarão erros são os seguintes:

- Desligamento de um equipamento, provocando um transiente na linha.
- Criação simultânea de mais de uma ficha durante o modo CF.
- Erros internos de software em algum equipamento.
- Perda da ficha, causada pelo desligamento do equipamento que possui a ficha no momento. Em todos os casos, o procedimento 'a' deve reestabelecer o funcionamento correto do sistema.

c) O equipamento que solicita uma aquisição ou transmissão para outro equipamento deve considerar a hipótese deste não responder. Neste caso, o solicitante dá um prazo para resposta (10 a 20ms em 9600 baud), após o qual a operação será considerada como não válida. Para evitar perda de tempo em caso de transferência de blocos longos, pode ser conveniente fazer uma inicialização antes para confirmar se o equipamento-alvo está disponível.

Resumo das operações básicas suportadas pelo PGC:

Operação Condições

Limpeza/Inicialização Solicitado pela linha
Leitura de porta I/O Solicitada pela linha
Escrita em porta I/O Solicitada pela linha
Leitura de memória Solicitada pela linha
Escrita em memória Solicitada pela linha
Leitura de bloco Solicitada pela linha
Escrita de bloco Solicitada pela linha
Assimilação de ficha Solicitada pela linha
Passagem de ficha Embutida no PGC
Transfere um bloco Invocada na aplicação (com ficha)
Adquire um bloco Invocada na aplicação (com ficha)
Transfere uma variável Invocada na aplicação (com ficha)
Adquire uma variável Invocada na aplicação (com ficha)

10 - Descrição do Protocolo B2P

O protocolo será descrito e implementado de acordo com o modelo ISO para interconexão de sistemas. Adotamos o modelo 2 do ISO-OSI, com três camadas (física, enlace, aplicação). Este é descrito pela sigla EPA, conforme a norma IEC870-5-3.

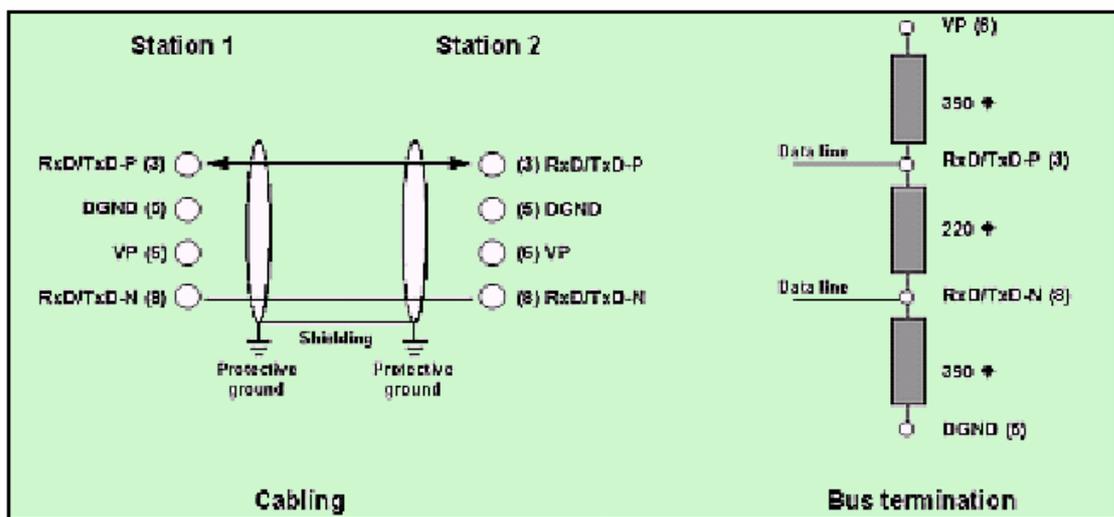
10.1 - Camada Física

10.1.1 - Interface Elétrica

Duas formas básicas estão previstas:

a) Padrão EIA RS232, para comunicações ponto a ponto, com cabo multivias blindado. Além dos sinais TxD e RxD poderão estar disponíveis os sinais de controle na forma prevista pela norma EIA, para os produtos que ofereçam o hardware necessário.

b) Padrão EIA RS485, para comunicação multiponto, com cabo de par trançado blindado. Além das recomendações previstas na norma, são adotadas terminações nas pontas da rede, na seguinte forma:



10.1.2 - Codificação dos Caracteres

Será usada transmissão assíncrona, com um bit de partida, 8 bits de dados, sem paridade 2 stop bits.

10.1.3 - Velocidades de Comunicação

São previstas as velocidades de 1,2k, 9,6k, 19,2k, 48k, 96k, 100k e 250 kbaud, selecionáveis nos equipamentos.

10.2 - Camada de Enlace

10.2.1 - Controle de Acesso ao Meio

As redes são estruturadas de duas formas básicas: Na rede com mestre (“unbalanced”, conforme IEC870), um equipamento central coordena o acesso ao meio e dele partem todas as mensagens de escrita ou leitura. Na rede sem mestre fixo (“balanced”, conforme IEC870), o controle de acesso ao meio é feito pelo método de passagem de ficha de forma circular entre todos os equipamentos ativos da rede.

10.2.2 - Serviços Básicos do Protocolo

Três tipos básicos de serviços estão definidos:

- Envio de mensagem, sem resposta esperada (S1 - Send/No Reply)
- Envio de mensagem, recebendo confirmação (S2 - Send/Confirm)
- Envio de mensagem, requisitando dados (S3 - Request/Respond)

O tipo de serviço que se aplica a cada operação está implícito na função e nos seus parâmetros, conforme definido na camada de aplicação.

10.2.3 - Estrutura Geral das Mensagens

Uma mensagem é definida como um pacote de informação que é produzido por um equipamento conectado à rede. Cada mensagem encapsula as informações referentes ao produtor da mensagem, ao destino desta, a função da mensagem e os dados correspondentes. São definidos meios para delimitação do início e do fim da mensagem e também da integridade dos dados que a constituem. O formato genérico para todas as mensagens é o seguinte:

INI	LEN	Dados da camada de aplicação	CRC
-----	-----	------------------------------	-----

INI	2 caracteres	Delimitador de início da mensagem (caracteres 05H e 64H)
LEN	1 caracter	Número de caracteres do campo de dados
Dados da camada de aplicação	1 a 251 caracteres	Campo de tamanho variável, usado pela camada de aplicação.
CRC	2 caracteres	Código de detecção de erros CRC-16, calculado sobre os dados da camada de aplicação, conforme o polinômio $X^{16}+X^{15}+X^2+1$. (Expresso na ordem LSB, MSB) (Este polinômio é o mesmo usado nos protocolos Modbus e Arcnet).

10.2.4 - Tratamento de Erros

O equipamento ligado à rede verifica que a linha está em silêncio. Duas condições são consideradas:

- Após um primeiro tempo configurável, o equipamento descarta a mensagem recebida até então.
- Após um segundo tempo configurável, o equipamento considera que a ficha foi perdida. Cada equipamento ativo possui a capacidade de recriar a ficha após um tempo de $(0,96 + X/25)$ s, onde X é o endereço do equipamento.

Observação: Caracteres recebidos com erros de overrun ou framing são descartados, e, portanto incluídos no período de silêncio.

O código de detecção de erro CRC calculado não coincide com o recebido. O equipamento deverá ignorar a parcela de mensagem já recebida. O sistema operacional dos CLPs oferece uma variável de atribuição fixa que comunica esta condição ao programa de alto nível.

Observação: Ocorrendo este erro na camada de enlace do equipamento origem ou destino, ele não executará a função nem enviará nenhuma resposta, uma vez que não se tem garantia da integridade da mensagem recebida.

10.3 - Camada de Aplicação

O protocolo da camada de aplicação define as diferentes funções de comunicação disponíveis e posteriormente com as instruções em alto nível se relacionam com elas. Para cada função é apresentada a estrutura de dados que é enviada à camada de enlace pelo par de equipamentos de dados que é enviada à camada de enlace pelo par de equipamentos envolvidos em cada operação de comunicação.

Origem: É o equipamento ativo - mestre ou equipamento de posse da ficha no momento.

Destino : Objetivo da operação, definido pelo equipamento origem.

As siglas usadas no detalhamento das funções:

FUN - Código da função a ser executada (1 caracter - Ver descrição a seguir).

END - Endereço do equipamento destino (1 ou mais caracteres – ver descrição a seguir).

ENDBLC - Endereço do bloco de memória para leitura ou escrita (2 caracteres na ordem LSB, MSB).

TAMBLC - Tamanho do bloco de memória para leitura (1 caracter máximo 253).

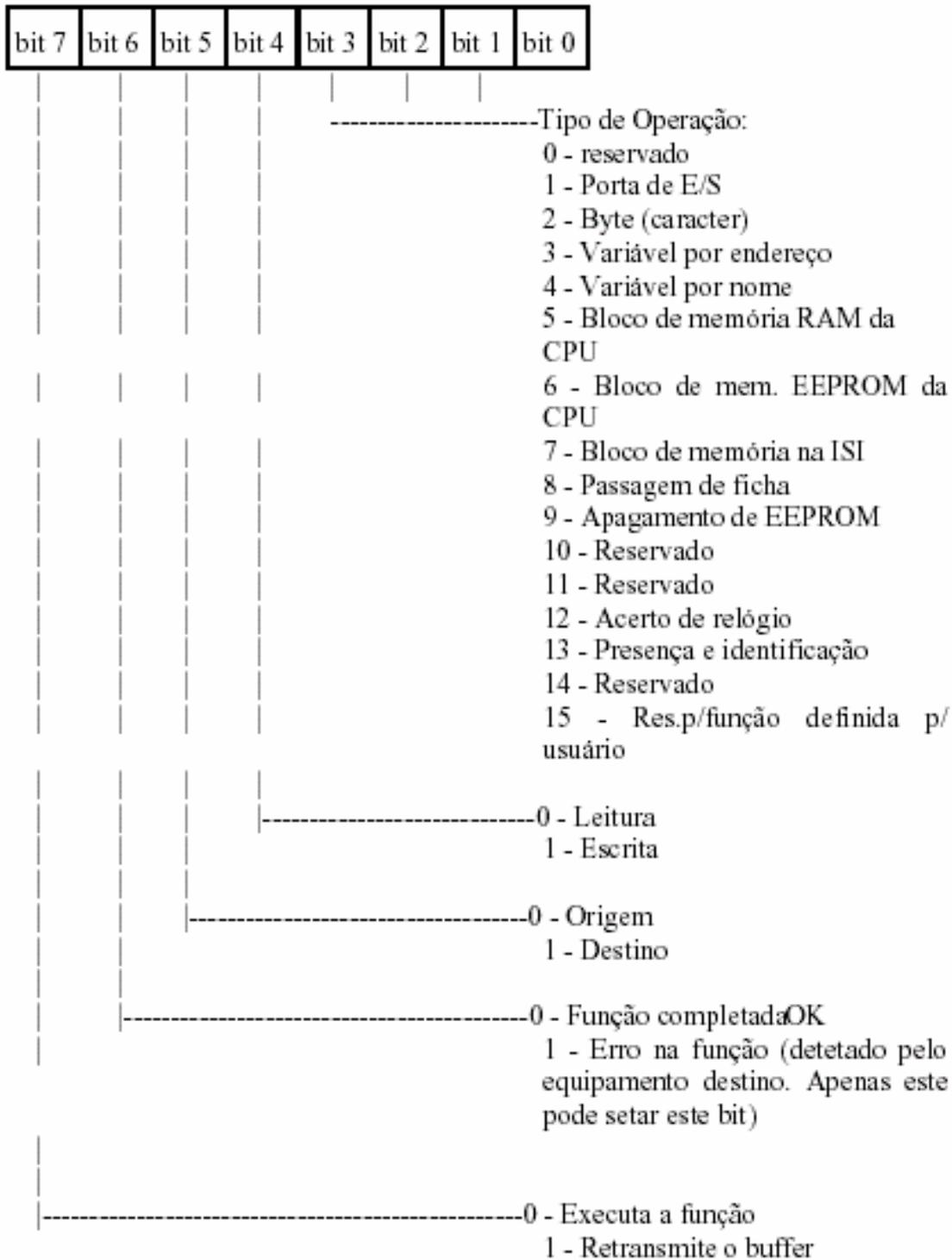
ENDVAR - Endereço do byte ou variável para leitura ou escrita (2 caracteres na ordem LSB, MSB).

ENDPOR - Endereço da porta de E/S para leitura ou escrita (1 caracter)

10.3.1 -Estrutura do Endereço de Destino

Faixa de endereços válidos: 1 a 126 (bits 0 a 6) O endereço 127 é usado para broadcasting. Uma função com esse endereço de destino é executada por todos os equipamentos da rede (válido para escrita de caracter, variável ou bloco e para acerto de relógio). Neste caso, os equipamentos destino não dão resposta. O bit 7 (quando 1) especifica endereços para retransmissão. É usado nos casos de rede com expansão não linear. Neste caso, são enviados vários caracteres de endereço: o primeiro, com o bit 7 alto é o endereço do primeiro retransmissor; o segundo, também com o bit 7 alto, é o endereço do segundo retransmissor e assim por diante. O último caracter, enviado sempre com o bit 7 baixo é o endereço do equipamento destino.

10.3.2 - Estrutura dos Códigos das Funções



10.3.4 - Tratamento de Erros

Todos os erros detectados pelo equipamento destino nesta camada provocam uma resposta em que o código de função é enviado com o bit 6 setado. O equipamento destino poderá marcar em um flag especial o tipo de erro que aconteceu. Nos CLPs, este flag é uma variável de atribuição fixa. As funções que envolvem leitura de dados são respondidas pelo equipamento destino com o campo “Dados solicitados” vazio. Alguns erros típicos são:

- Função inexistente ou não suportada pelo equipamento destino.
- Endereço de memória ou porta de E/S inexistente ou protegido para escrita ou leitura.
- Resposta negativa nas funções de verificação (por exemplo: Eprom apagada).

10.3.5 - Programação em Alto Nível

Nos CLPs, o programa em alto nível irá selecionar o tipo de protocolo através do comando de configuração, onde é configurado o protocolo, taxa de transmissão, endereço lógico, etc.

Esta parte sinaliza para o Sistema Operacional qual o canal sobre o qual irão operar as instruções LE e ESCREVE. As instruções LE e ESCREVE são usadas da mesma forma, independente do tipo de protocolo adotado em cada aplicação. Além das instruções, o sistema operacional oferece variáveis de atribuição fixa para que o programa do usuário possa saber o que está acontecendo na rede.

11. Projeto do Sistema e Programação do CLP para Aplicações em Rede

O equipamento principal de controle dentro da rede local é o Controlador Programável. Cabe a ele a relação direta do sistema de automação com os elementos de campo: Sensores e atuadores. O CLP é relacionado a estes elementos. De um ponto de vista horizontal, uma linha de produção ou uma planta de processo é atendida por vários controladores interligados. Do ponto de vista vertical, a rede local é o fundamento para a conexão com computadores de processo, unidades centrais e sistemas integrados de supervisão. A comunicação do CLP com os outros elementos da rede pode ser considerada de duas formas: Passiva ou Ativa.

- Na forma passiva, o CLP não executa nenhuma instrução específica com relação a rede. São os outros equipamentos que lêem e escrevem valores na memória; o CLP apenas atende as rotinas normais de controle usando estes valores.
- Na forma ativa, instruções específicas de programação são usadas para ler e escrever valores ou blocos de dados em outros equipamentos. Estes recursos serão especialmente detalhados a seguir.

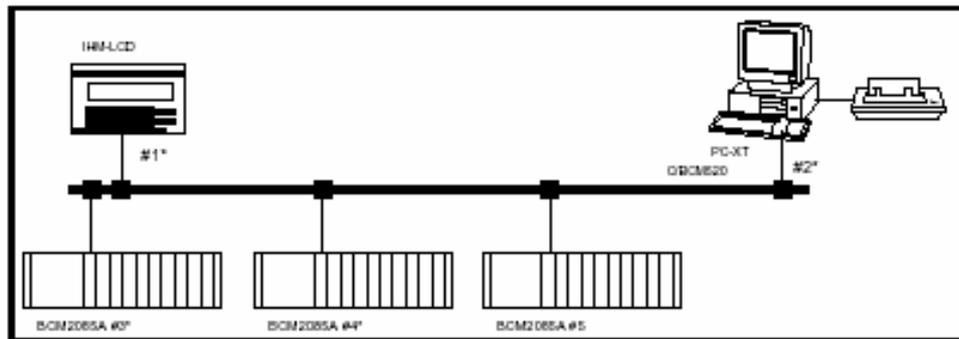
Observe que as técnicas não são excludentes - Um CLP pode trabalhar com comunicação ativa e passiva na mesma aplicação, em momentos diferentes.

O projeto da parte de comunicação de uma aplicação de CLPs deve começar pela definição exata do intercâmbio de informações - isto é, que dados cada equipamento precisa? Onde estão disponíveis? Estes dados serão buscados ou transmitidos?

Com a definição do intercâmbio pronta, passamos à montagem física e lógica da rede. A figura abaixo mostra um exemplo simples, onde já está implícita a solução às seguintes questões:

- Quais os equipamentos que serão interligados?
- Quais equipamentos executarão tarefas ativas de comunicação?
- Qual o endereço lógico de cada equipamento na rede?

Exemplo de aplicação



A rede deve ser projetada de modo que a falha em um dos componentes não cause a paralisação total do sistema. Os equipamentos devem dispor de procedimentos de autoteste, de modo que a qualquer instante seja possível verificar o seu correto funcionamento. Estas considerações relativas à segurança devem ter particular importância no projeto do fluxo de dados, fontes de alimentação, etc.

O passo seguinte - Programação dos CLPs, deve levar em conta o conjunto de definições anteriores, além da especificação relativa às suas próprias tarefas de controle. As Linguagens de programação em alto nível possui uma série de recursos para tratar das tarefas relativas à comunicação com a rede.

12. Desempenho das instalações

Neste item consideraremos dois aspectos relacionados ao desempenho das instalações: Tempos de comunicação e Detecção de erros.

12.1 - Tempos de comunicação

Um dos aspectos de projeto mais importantes relacionados a um sistema de automação distribuído é o seguinte: Quanto tempo um determinado equipamento leva para produzir uma resposta, em função de um evento ocorrido em outro equipamento? Descontados os aspectos internos de aquisição, processamento e atuação de cada

equipamento, iremos calcular qual o prazo máximo de cumprimento de uma tarefa de comunicação.

TCIRC	Tempo total (em segundos) de circulação da ficha na rede. Corresponde ao prazo máximo de cumprimento de uma tarefa de comunicação.	TTk	Tempo de transferência da mensagem completa referente ao equipamento 'k'.
TPR	Tempo médio de processamento das operações de comunicação. tipicamente de 100 us.	NCH	Número de caracteres da mensagem.
TFC	Tempo de passagem e aceitação da ficha de um equipamento para o seguinte.	Br	Taxa de transmissão (velocidade em bauds).
Tsk	Tempo de transmissão da solicitação de uma tarefa de comunicação pelo equipamento 'k'.	N	Número de equipamentos da rede.

O valor de TCIRC é dado pela fórmula:

$$TCIRC = N(TPR + TFC) + \sum_{k=1}^N Tsk + TTk$$

onde:

- TFC = 44/Br

- Tsk ==> Depende da tarefa solicitada:

88/Br para leitura/escrita de variável

132/Br para leitura/escrita de bloco

- Ttk = 22.NCH/Br

Além do procedimento de cálculo descrito acima, o tempo total de circulação da ficha na rede pode ser avaliado na prática, em tempo real, pela variável CFIC. Esta variável de atribuição fixa é incrementada a cada passagem de ficha pelo CLP.

12.2 - Detecção de erros

A estratégia para garantia da isenção de erros em um sistema de controle distribuído deve combinar os recursos automáticos dos sistemas operacionais dos CLP's com rotinas pré-determinadas incluídas nos programas aplicativos dos equipamentos. Os erros de comunicação na rede são causados principalmente por:

a) Introdução de ruído elétrico no meio físico (cabo), devido ao uso de um cabo inadequado, passagem junto a fontes externas de ruído ou falta de terminação.

b) Falhas elétricas na instalação causadas por mau contato nas ligações, emendas, fontes com tensão incorreta, etc.

c) Desligamento ou defeito intermitente em algum dos equipamentos da rede.

d) Rotinas assembler do usuário que travam, mudam parâmetros ou interferem de forma crítica nas rotinas do sistema operacional responsáveis pelo tratamento da comunicação. Independentemente da causa da falha, os erros de comunicação podem aparecer das seguintes formas:

1) Perda da ficha, exigindo a sua recriação.

2) Operações iniciadas e não completadas devido a erros detectados na transmissão ou na recepção.

3) Operações completadas sem erros detectados, mas que produzem resultados incorretos.

Os erros dos tipos '1' e '2' podem ser monitorados através das variáveis de atribuição fixa específicas.

Os erros do tipo '3' tem uma probabilidade de ocorrência muito menor; No entanto em aplicações críticas é necessário incluir algum processo extra de verificação em alto nível. Algumas formas usualmente adotadas são:

Repetição da mensagem

Inclusão de um checksum

Pedido de confirmação

Outros

13. Geração do código de CRC (Checagem de Redundância Cíclica)

O Campo de *Checagem de Redundância Cíclica* são dois bytes, contendo um valor binário de 16 bits. O Valor de CRC é calculado pelo equipamento emissor na formação do *frame*.

O dispositivo receptor recalcula o CRC durante a recepção do frame e compara com o valor calculado com o valor recebido no campo CRC. Se os valores forem iguais o *frame* chegou sem erro, caso contrário o sistema operacional descarta o *frame* recebido.

Para calcular o valor do CRC se carrega um registro de 16 bits todos com 1. Depois começa um processo que pega os bytes sucessivos do *frame* e realiza operação com o conteúdo do registro, atualizando este com o resultado obtido.

Só os 8 bits de dados de cada caracter são utilizados para gerar o CRC, os bits de *start* e *stop* não interferem na criação do CRC.

Durante a geração do CRC, efetua-se uma operação booleana XOR a cada 8 bits com o conteúdo do registro. Depois o resultado se aplica a um deslocador de bit na direção do bit menos significativo(LSB).

O LSB é extraído e examinado, se o LSB for 1 se realiza uma operação XOR entre o registro e um outro valor fixo preestabelecido, se o LSB for 0 não se efetua XOR.

Este processo é repetido até ser cumprido 8 deslocamentos de bits. Depois do ultimo deslocamento (oitavo bit) o próximo byte é operado com XOR com o valor atual do registro e o processo se repete com os oito deslocamentos mais, como esta descrito acima com todos os bytes do *frame*.

13.1 Procedimento para gerar o CRC16:

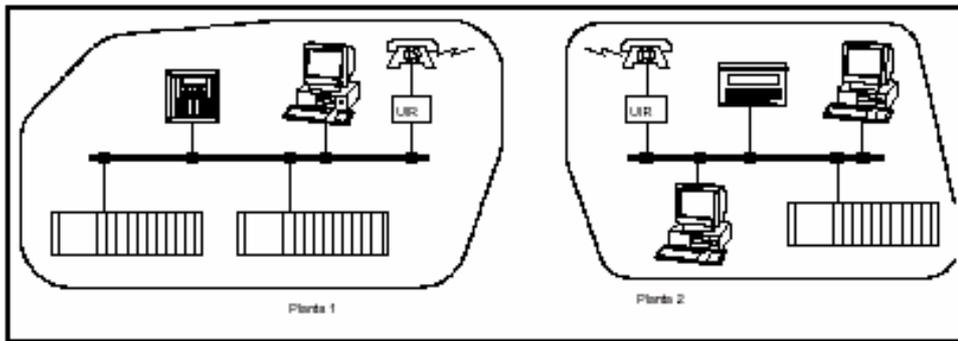
1. Carregar um registro de 16 bits que denominaremos registro CRC, com FFFF (todos 1).
2. Fazer XOR do primeiro byte - 8 bits – do frame com o byte de baixa ordem do registro CRC de 16 bits, colocando o resultado no registro CRC.
3. Deslocar o registro CRC de um bit a direita, extrair e examinar o LSB.
4. (Se o LSB era 0): Repetir passo 3 (outro deslocamento). (Se o LSB era 1): Fazer XOR entre o registro CRC e o valor A001 hex (1010 0000 0000 0001).
5. Repetir os passos 3 e 4 até que se efetue 8 deslocamentos. Uma vez feito isto, o processo foi concluído com um byte completo – 8 bits -.
6. Repetir os passos 2 ao 5 para o próximo byte – 8 bits – do frame. Continuar fazendo isto, até que todos os bytes tenham sido processados.
7. O conteúdo final do registro CRC é o valor CRC.
8. O CRC situado na mensagem, tem seus bytes de ordem alto e baixo invertidos.

14 - Extensão para redes de longa distância

O uso de redes de longa distância (WAN ou Wide Area Network) estende as potencialidades dos sistemas de controle distribuído para estruturas cujos componentes estão geograficamente dispersos. Genericamente, temos duas ou mais redes locais interligadas por algum tipo de telecomunicação (linha telefônica discada, LPCD ou rádio

enlace). Os equipamentos estão espalhados pelas duas áreas e devemos prover um meio de transmissão de mensagens entre todos eles.

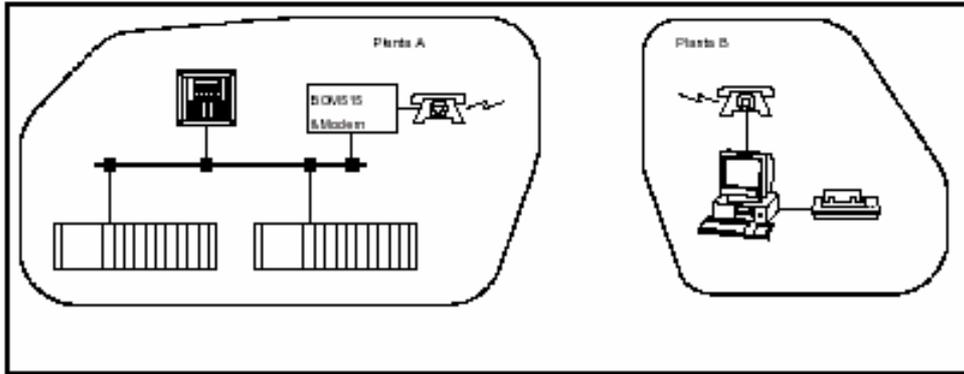
Exemplo de rede de longa distância.



A introdução dos recursos de telecomunicação introduz alguns fatores novos que devem ser considerados na concepção do sistema distribuído:

- a) A distância física entre as plantas pode ser bastante considerável - Não podemos mais desprezar o tempo de propagação dos sinais.
- b) Os equipamentos de telecomunicação possuem alguma forma de processamento e tratamento do sinal; isto provoca atrasos significativos nas comunicações.
- c) Devido aos tempos 'a' e 'b' e a natureza da telecomunicação, ocorreriam ecos de retransmissão na rede se não houver um tratamento especial do sinal.
- d) No caso de linhas telefônicas discadas, a comunicação entre as plantas não é permanente. A inclusão da Unidade de Interface com Rede (UIR ou NIU, de "Network Interface Unit") resolve estes problemas através do pré e pós-processamento dos sinais que trafegam pelos equipamentos de telecomunicação. O equipamento UIR deve ser configurado de acordo com a topologia e endereços lógicos das estações. Uma vez em funcionamento, as estações tem a capacidade de comunicação integral dentro do protocolo PGC, tornando transparente a existência da telecomunicação. Um caso especial que dispensa o uso do equipamento UIR é apresentado na figura a seguir. Existe uma rede de equipamentos na planta 'A' e um computador de supervisão/operação na planta B.

Caso especial de rede de longa distância



Nesta aplicação, existe apenas um computador de processo na planta 'B', ligado diretamente à linha telefônica. Não existe nenhum equipamento da norma RS485 na planta 'B' - Todos os controladores programáveis estão localizados na planta 'A'. O software do computador implementa o protocolo PGC, com circulação de ficha e troca de mensagens entre todos os quatro equipamentos. No software do computador, a rotina de comunicação foi adaptada para tratar o eco de retransmissão e impedir que este cause erros no sistema.