

MINICURSO

Comunicação Serial - RS232

Edmur Canzian
CNZ Engenharia e Informática Ltda.
<http://www.cnz.com.br>
e-mail: engenharia@cnz.com.br

Conceitos Básicos sobre Comunicação Serial

O que é 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.

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.

Comunicação Serial

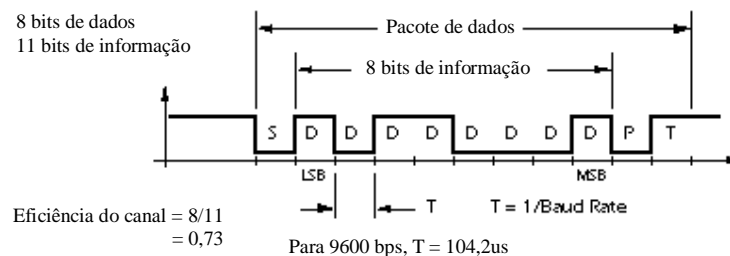
A maioria das mensagens digitais são mais longas que alguns poucos bits. Por não ser prático nem econômico transferir todos os bits de uma mensagem simultaneamente, a mensagem é quebrada em partes menores e transmitida seqüencialmente. A transmissão bit-serial converte a mensagem em um bit por vez através 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.

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 μ s (1/9600 s).

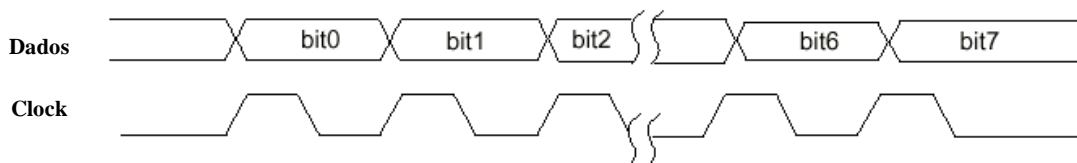
Outro conceito é a eficiência do canal de comunicação que é definido como o número de bits de informação utilizável (dados) enviados através do canal por segundo. Ele não inclui bits de sincronismo, formatação, e detecção de erro que podem ser adicionados a informação antes da mensagem ser transmitida,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



Transmissão Assíncrona x Transmissão Síncrona

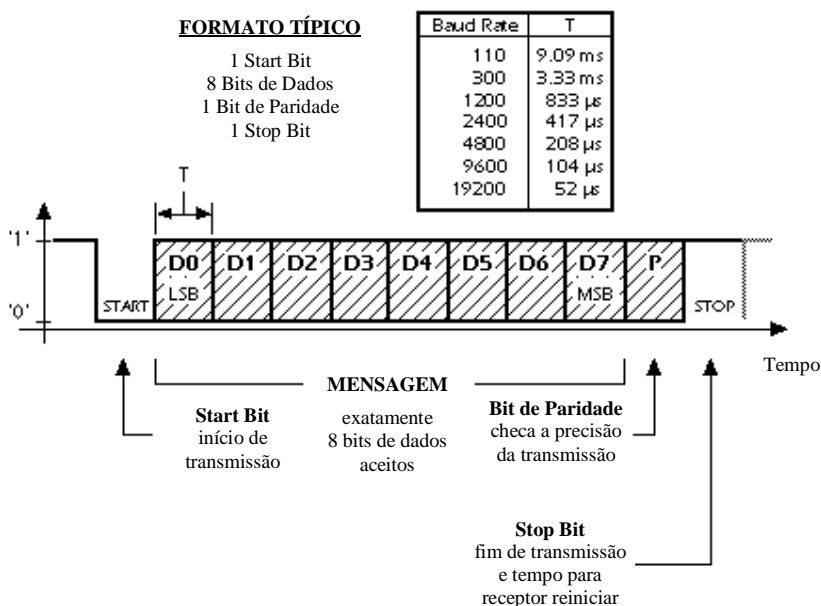
Geralmente, dados serializados não são enviados de maneira uniforme através de um canal. Ao invés disso, pacotes com informação regulares são enviados seguidos de uma pausa. Os pacotes de dados binários são enviados dessa maneira, possivelmente com comprimentos de pausa variável entre pacotes, até que 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").



O comprimento do pacote de dados é pequeno em sistemas assíncronos para minimizar o risco do oscilador do transmissor e do receptor variar. Quando osciladores a cristal são utilizados, a sincronização pode ser garantida sobre os 11 bits de período. A cada novo pacote enviado, o “start bit” reseta a sincronização, portanto a pausa entre pacotes pode ser longa.

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.

HEX	DEC	CHR	HEX	DEC	CHR	HEX	DEC	CHR	HEX	DEC	CHR
00	00	NUL	20	32	SPC	40	64	@	60	96	`
01	01	SOH	21	33	!	41	65	A	61	97	a
02	02	STX	22	34	"	42	66	B	62	98	b
03	03	ETX	23	35	#	43	67	C	63	99	c
04	04	EOT	24	36	\$	44	68	D	64	100	d
05	05	ENQ	25	37	%	45	69	E	65	101	e
06	06	ACK	26	38	&	46	70	F	66	102	f
07	07	BEL	27	39	'	47	71	G	67	103	g
08	08	BS	28	40	(48	72	H	68	104	h
09	09	HT	29	41)	49	73	I	69	105	i
0A	10	LF	2A	42	*	4A	74	J	6A	106	j
0B	11	VT	2B	43	+	4B	75	K	6B	107	k
0C	12	FF	2C	44	,	4C	76	L	6C	108	l
0D	13	CR	2D	45	-	4D	77	M	6D	109	m
0E	14	SO	2E	46	.	4E	78	N	6E	110	n
0F	15	SI	2F	47	/	4F	79	O	6F	111	o
10	16	DLE	30	48	0	50	80	P	70	112	p
11	17	DC1	31	49	1	51	81	Q	71	113	q
12	18	DC2	32	50	2	52	82	R	72	114	r
13	19	DC3	33	51	3	53	83	S	73	115	s
14	20	DC4	34	52	4	54	84	T	74	116	t
15	21	NAK	35	53	5	55	85	U	75	117	u
16	22	SYN	36	54	6	56	86	V	76	118	v
17	23	ETB	37	55	7	57	87	W	77	119	w
18	24	CAN	38	56	8	58	88	X	78	120	x
19	25	EM	39	57	9	59	89	Y	79	121	y
1A	26	SUB	3A	58	:	5A	90	Z	7A	122	z
1B	27	ESC	3B	59	;	5B	91	[7B	123	{
1C	28	FS	3C	60	<	5C	92	\	7C	124	
1D	29	GS	3D	61	=	5D	93]	7D	125	}
1E	30	RS	3E	62	>	5E	94	^	7E	126	~
1F	31	US	3F	63	?	5F	95	_	7F	127	DEL

Os caracteres não imprimíveis (00 a 31) são utilizados por diversos protocolos de comunicação. A nomenclatura dada para cada um deles é a seguinte:

DEC	CHR	CTRL-x	NOME
00	NUL	CTRL-@	Null
01	SOH	CTRL-A	Start of Heading
02	STX	CTRL-B	Start of Text
03	ETX	CTRL-C	End of Text
04	EOT	CTRL-D	End of Transmition
05	ENQ	CTRL-E	Enquiry
06	ACK	CTRL-F	Acknowledge
07	BEL	CTRL-G	Bell
08	BS	CTRL-H	Backspace
09	HT	CTRL-I	Horizontal Tab
10	LF	CTRL-J	Line Feed
11	VT	CTRL-K	Vertical Tab
12	FF	CTRL-L	Form Feed
13	CR	CTRL-M	Carriage Return
14	SO	CTRL-N	Shift Out
15	SI	CTRL-O	Shift In
16	DLE	CTRL-P	Data Line Escape
17	DC1	CTRL-Q	Device Control 1
18	DC2	CTRL-R	Device Control 2
19	DC3	CTRL-S	Device Control 3
20	DC4	CTRL-T	Device Control 4
21	NAK	CTRL-U	Not Acknowledge
22	SYN	CTRL-V	Synchronous
23	ETB	CTRL-W	End of Transmition Block
24	CAN	CTRL-X	Cancel
25	EM	CTRL-Y	End of Medium
26	SUB	CTRL-Z	Substitute
27	ESC	CTRL-[Escape
28	FS	CTRL-\	File Separator
29	GS	CTRL-]	Group Separator
30	RS	CTRL-^	Record Separator
31	US	CTRL-_	Unit Separator

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 para 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.

$$\begin{array}{r}
 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1 \\
 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0 \\
 +\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ \text{Dados} \\
 \hline
 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \\
 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \\
 \hline
 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ \text{Soma Aritmética} \\
 \\
 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ \text{Soma truncada – 8 bits} \\
 +\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ \text{Checksum (complemento de 2)} \\
 \hline
 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \text{Soma + Checksum = 0}
 \end{array}$$

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.

Interface Serial RS232 (EIA232)¹

O que é RS232 ?

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 consequentemente 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:

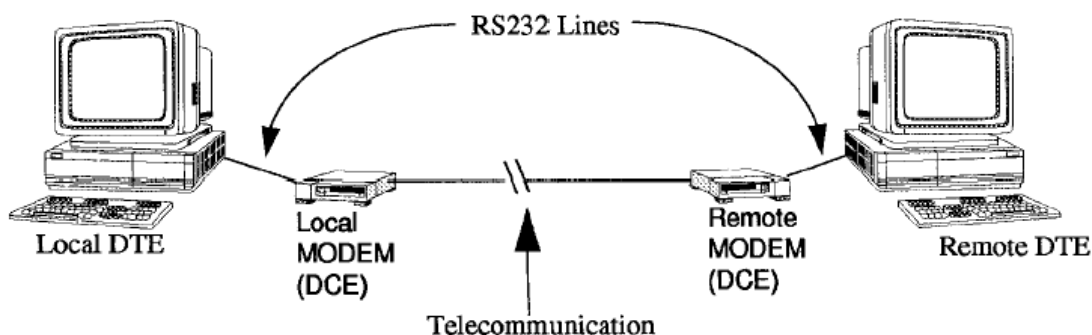
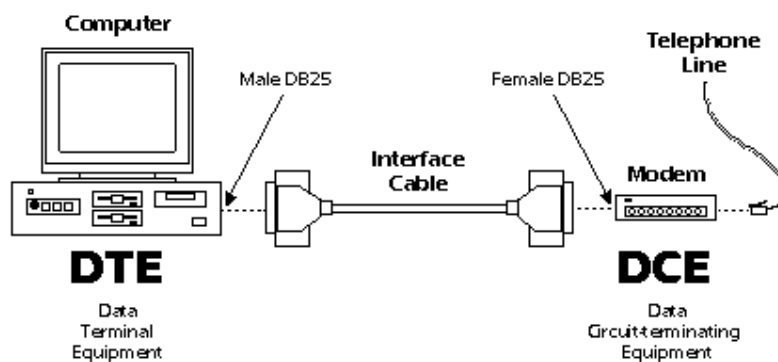
- ✓ A ausência ou conexão errada de sinais de controle, resultam em estouro do buffer (“overflow”) ou travamento da comunicação.
- ✓ Função incorreta de comunicação para o cabo em uso, resultam em inversão das linhas de Transmissão e Recepção, bem como a inversão de uma ou mais linhas de controle (“handshaking”).

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

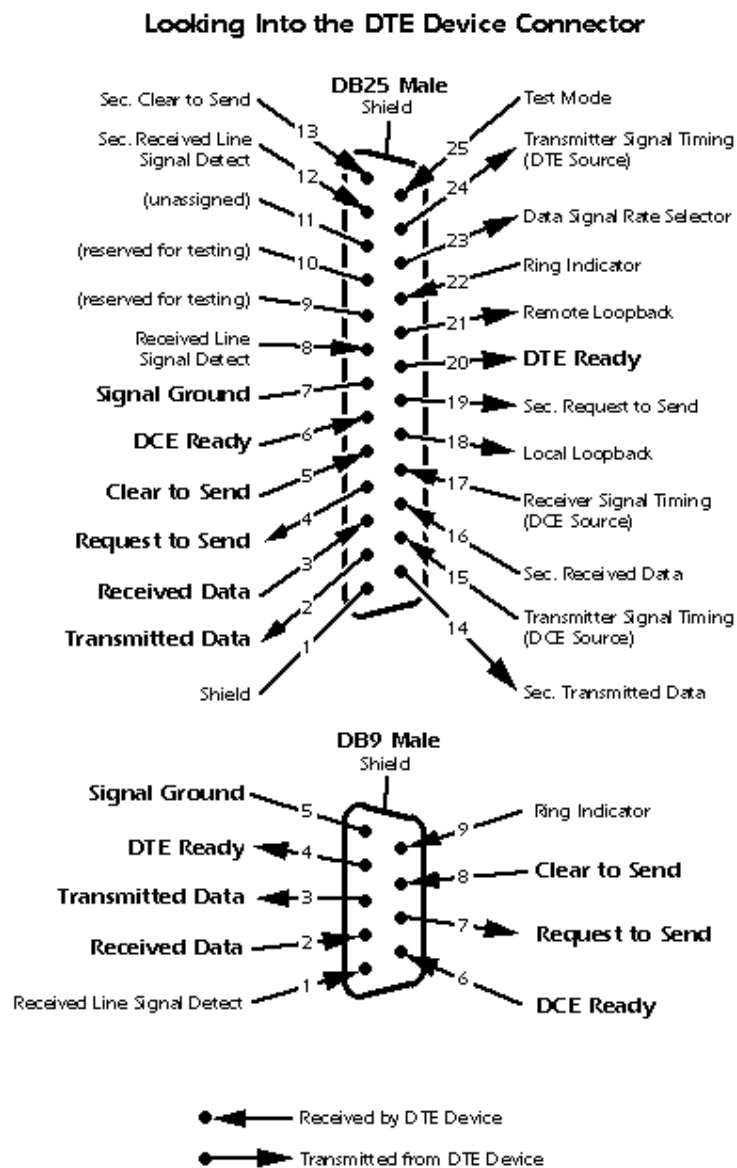
¹ O termo RS232 será utilizado nesta apostila quando o texto fizer referências à interface de comunicação. O termo EIA232 será utilizado quando o texto fizer referências à norma estabelecida pela EIA.

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.

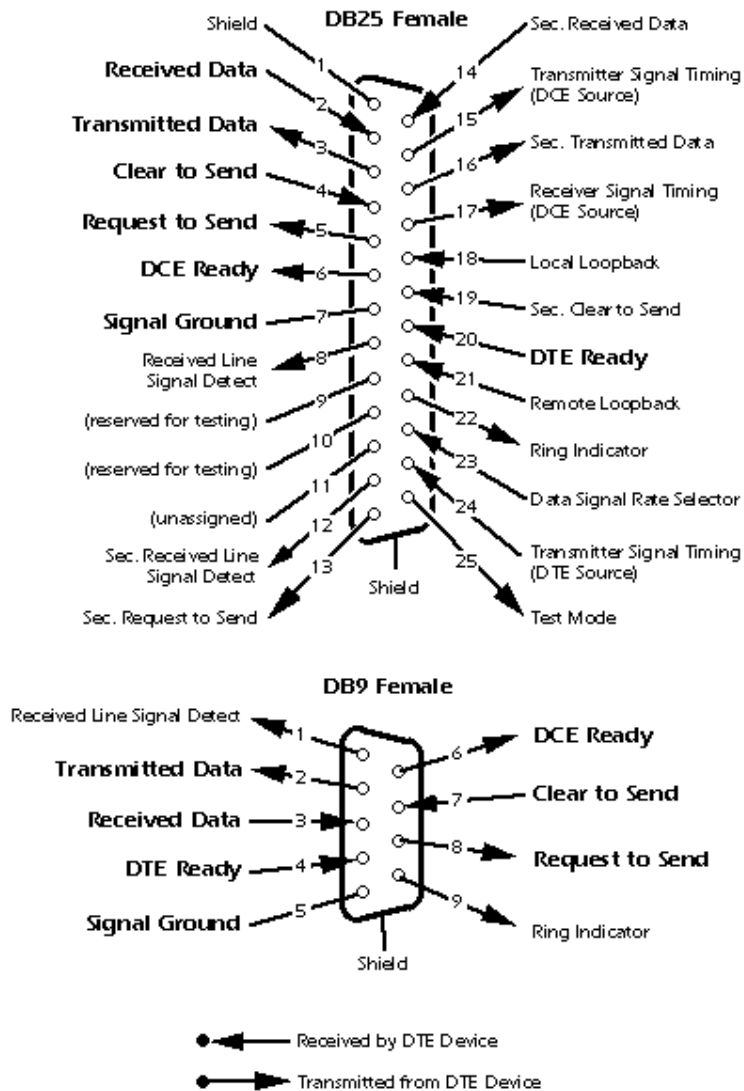


Na figura a seguir é apresentada a definição dos sinais para um dispositivo DTE (usualmente um micro PC). Os sinais mais comuns são apresentados em negrito.



Na figura a seguir é apresentada a definição dos sinais para um dispositivo DCE (usualmente um modem). Os sinais mais comuns são apresentados em **negrito**.

Looking Into the DCE Device Connector

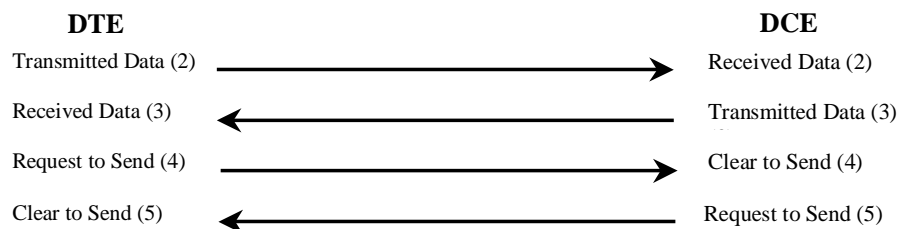


Diversos sinais são necessários para conexões onde o dispositivo DCE é um modem, e eles são utilizados apenas quando o protocolo de software os emprega. Para dispositivos DCE que não são modem, ou quando dois dispositivos DTE são conectados diretamente, poucos sinais são necessários.

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.



Pinagem

As funções dos sinais da norma EIA232 podem ser subdivididos em 6 categoria, apresentados na tabela a seguir.

Pino	Nome	Descrição
<i>Sinais de Terra</i>		
1	Shield	Sinal de terra de proteção (malha de aterramento do cabo e carcaça do conector).
7	Ground (GND)	Sinal de terra utilizado como referência para outros sinais.
<i>Canal de Comunicação Primário</i>		
2	Transmitted Data (TxD)	Este sinal está ativo quando dados estiverem sendo transmitidos do DTE para o DCE. Quando nenhum dado estiver sendo transmitido, o sinal é mantido na condição de marca (nível lógico “1”, tensão negativa).
3	Received Data (RxD)	Este sinal está ativo quando o DTE receber dados do DCE. Quando o DCE estiver em repouso, o sinal é mantido na condição de marca (nível lógico “1”, tensão negativa).
4	Request To Send (RTS)	Este sinal é habilitado (nível lógico “0”) para preparar o DCE para aceitar dados transmitidos pelo DTE. Esta preparação inclui a habilitação dos circuitos de recepção, ou a seleção a direção do canal em aplicações half-duplex. Quando o DCE estiver pronto, ele responde habilitando o sinal CTS
5	Clear To Send (CTS)	Este sinal é habilitado (nível lógico “0”) pelo DCE para informar ao DTE que a transmissão pode começar. Os sinais RTS e CTS são comumente utilizados no controle do fluxo de dados em dispositivos DCE.

Pino	Nome	Descrição
Canal de Comunicação Secundário		
14	Secondary Transmitted Data (STxD)	Equivalente ao sinal TxD, porém válido para o canal secundário.
16	Secondary Received Data (SRxD)	Equivalente ao sinal RxD, porém válido para o canal secundário.
19	Secondary Request To Send (SRTS)	Equivalente ao sinal RTS, porém válido para o canal secundário.
13	Secondary Clear To Send (SCTS)	Equivalente ao sinal CTS, porém válido para o canal secundário.
Sinais de Controle e de Status de Modem		
6	DCE Ready (DSR)	<p>Também chamado de Data Set Ready. Quando originado de um modem, este sinal é habilitado (nível lógico “0”) quando as seguintes forem satisfeitas:</p> <ul style="list-style-type: none"> 1 - O modem estiver conectado a uma linha telefônica ativa e “fora do gancho; 2 - O modem estiver no modo dados; 3 – O modem tiver completado a discagem e está gerando um tom de resposta. <p>Se a linha for tirada do gancho, uma condição de falha for detectada, ou uma conexão de voz for estabelecida, o sinal DSR é desabilitado (nível lógico “1”).</p>
20	DTE Ready (DTR)	<p>Também chamado de Data Terminal Ready. Este sinal é habilitado (nível lógico “0”) pelo DTE quando for necessário abrir o canal de comunicação. Se o DCE for um modem, a habilitação do sinal DTR prepara o modem para ser conectado ao circuito do telefone, e uma vez conectado, mantém a conexão. Quando o sinal DTR for desabilitado (nível lógico “1”), o modem muda para a condição “no gancho” e termina a conexão.</p>
8	Received Line Signal Detector (CD)	<p>Também chamado de Data Carrier Detect (DCD). Este sinal é relevante quando o DCE for um modem. Ele é habilitado (nível lógico “0”) quando a linha telefônica está “fora do gancho”, uma conexão for estabelecida, e um tom de resposta começar a ser recebido do modem remoto. Este sinal é desabilitado (nível lógico “1”) quando não houver tom de resposta sendo recebido, ou quando o tom de resposta for de qualidade inadequada para o modem local.</p>
12	Secondary Received Line Signal Detector (SCD)	Este sinal é equivalente ao CD, porém refere-se ao canal de comunicação secundário.
22	Ring Indicator (RI)	<p>Este sinal é relevante quando o DCE for um modem, e é habilitado (nível lógico “0”) quando um sinal de chamada estiver sendo recebido na linha telefônica. A habilitação desse sinal terá aproximadamente a duração do tom de chamada, e será desabilitado entre os tons ou quando não houver tom de chamada presente.</p>
23	Data Signal Rate Selector	<p>Este sinal pode ser originado tanto no DTE quanto no DCE (mas não em ambos), e é usado para selecionar um de dois “baud rates” pré-configurados. Na condição de habilitação (nível lógico “0”) o “baud rate” mais alto é selecionado.</p>

Pino	Nome	Descrição
<i>Sinais de Transmissão e Recepção de Tempos</i>		
15	Transmitter Signal Element Timing (TC)	Também chamado de Transmitter Clock (TxC). Este sinal é relevante apenas quando o DCE for um modem e operar com um protocolo síncrono. O modem gera este sinal de clock para controlar exatamente a taxa na qual os dados estão sendo enviado pelo pino TxD, do DTE para o DCE. A transição de um nível lógico “1” para nível lógico “0” nessa linha causa uma transição correspondente para o próximo bit de dado na linha TxD.
17	Receiver Signal Element Timing (RC)	Também chamado de Receiver Clock (RxC). Este sinal é similar ao sinal TC descrito acima, exceto que ele fornece informações de temporização para o receptor do DTE.
24	Transmitter Signal Element Timing (ETC)	Também chamado de External Transmitter Clock. Os sinais de temporização são fornecidos externamente pelo DTE para o uso por um modem. Este sinal é utilizado apenas quando os sinais TC e RC não estão sendo utilizados.
<i>Sinais de Teste do Canal de Comunicação</i>		
18	Local Loopback (LL)	Este sinal é gerado pelo DTE e é usado para colocar o modem no estado de teste. Quando o sinal LL for habilitado (nível lógico “0”), o modem redireciona o sinal de saída modulado, que normalmente vai para o linha telefônica, de volta para o circuito de recepção. Isto habilita a geração de dados pelo DTE serem ecoados através do próprio modem. O modem habita os sinal TM reconhecendo que ele está na condição de “loopback”.
21	Remote Loopback (RL)	Este sinal é gerado pelo DTE e é usado para colocar o modem remoto no estado de teste. Quando o sinal RL é habilitado (nível lógico “0”), o modem remoto redireciona seus dados recebidos para a entrada, voltando para o modem local. Quando o DTE inicia esse teste, o dado transmitido passa através do modem local, da linha telefônica, do modem remoto, e volta, para exercitar o canal e confirmar sua integridade.
25	Test Mode (TM)	Este sinal é relevante apenas quando o DCE é um modem. Quando habilitado (nível lógico “0”), indica que o modem está em condição de teste local (LL) ou remoto (RL).

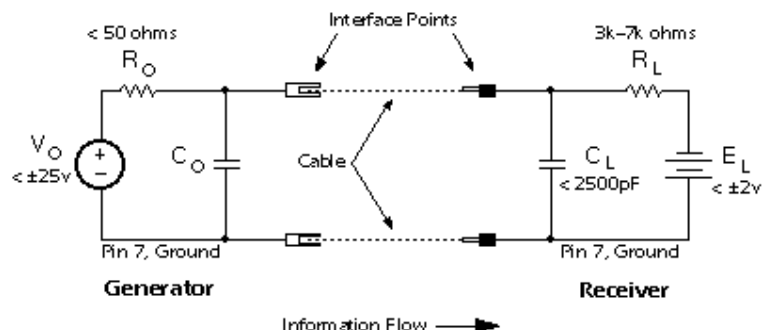
Sinal de Terra Comum

A norma EIA232 inclui a referência de terra no Pino 7, e é freqüentemente conectada ao Pino 1 e a blindagem do cabo que envolve os demais condutores. Sinais de tensão dos dados, temporizações e controle são medidos com relação a esse terra comum. Equipamentos que utilizam a interface RS232 não podem ser utilizados em aplicações onde o equipamento nos dois opostos devem estar eletricamente isolados.

Isoladores ópticos podem ser usados para garantir isolamento, contudo, isso não é mencionado ou incluído na especificação da norma EIA232.

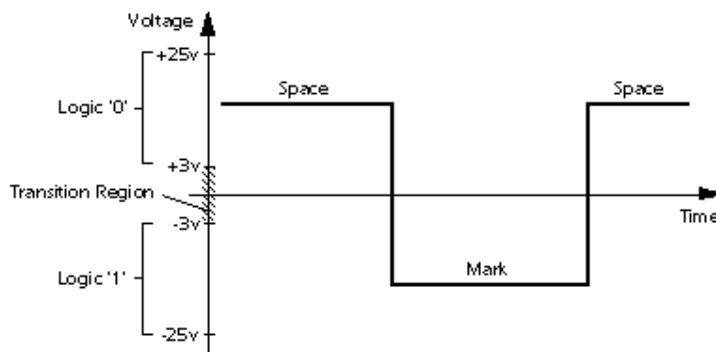
Características dos Sinais

Todas as linhas, sejam elas de informações de dados, temporização ou controle, podem ser representadas pelo mesmo circuito elétrico equivalente da figura a seguir:



Este circuito equivalente aplica-se aos sinais originados tanto no DTE quanto no DCE. A capacitância “Co” não é especificada na norma, mas deve ser assumida como pequena e consistir apenas de elementos parasitas. “Ro” e “Vo” são escolhidos de forma tal que a corrente de curto-circuito não exceda a 500 mA.

Sinais com tensão entre -3 volts e +3 volts com relação ao terra (pino 7) são considerados nível lógico “1” (condição marca), e tensões entre +3 volts e +25 volts são considerados nível lógico “0” (condição espaço). A faixa de tensões entre -3 volts e +3 volts é considerada uma região de transição para o qual o estado do sinal é indefinido.



IMPORTANTE: Se forem inseridos LEDs ou circuitos de teste para visualizar o estado dos sinais, o sinal de tensão cairá em magnitude e poderá afetar o rendimento da interface se o cabo for longo. Também deve-se notar que alguns periféricos baratos são alimentados com os próprios sinais da interface para não utilizar fonte de alimentação própria. Embora isso normalmente funcione sem problemas, mantenha o cabo curto, e tome cuidado que a imunidade a ruídos irá diminuir.

Os sinais de saída foram projetados para funcionar em aberto, ou com curto-circuito com outros sinais do condutor, incluindo o sinal de terra, sem danificar o outro circuito associado. Os sinais de entrada também foram projetados para aceitar qualquer tensão entre ± 25 volts sem danificar.

Quatro sinais foram implementados com segurança à falhas (“fail-safe design”) no qual durante a desenergização ou desconexão do cabo, seus sinais estarão desabilitados (nível lógico “0”). São eles:

- ✓ Sinal RTS – desabilitado
- ✓ Sinal SRTS – desabilitado
- ✓ Sinal DTR – DTE não pronto
- ✓ Sinal DSR – DCE não pronto

Temporização dos Sinais

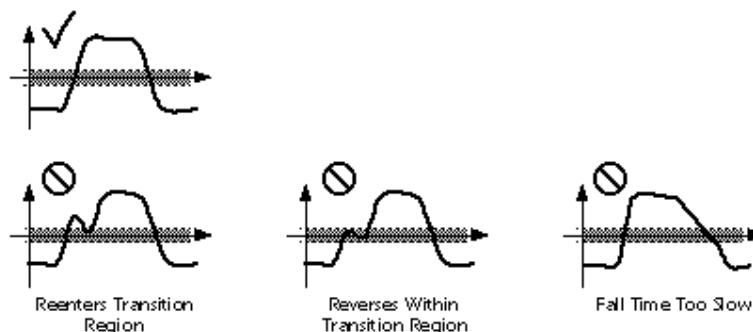
A norma EIA232 especifica uma taxa máxima de transferência de dados de 20.000 bits por segundo (o limite usual é 19200 bps). Baud rates fixos não são fornecidos pela norma. Contudo, os valores comumente usados são 300, 1200, 2400, 4800, 9600 e 19200 bps.

Mudanças no estado dos sinais de nível lógico “1” para “0” ou vice-versa devem seguir diversas características, dadas a seguir:

- ✓ Sinais que entram na zona de transição durante uma mudança de estado deve atravessar essa região com direção ao estado oposto sem reverter a direção ou reentrar;
- ✓ Para os sinais de controle, o tempo na zona de transição deve ser menor do que 1 ms;

Para sinais de temporização, o tempo para atravessar a zona de transição deve ser:

- ✓ Menor do que 1 ms para períodos de bits maiores que 25 ms;
- ✓ 4% do período de um bit para períodos entre 25 ms e 125 μ s;
- ✓ Menor do que 5 μ s para períodos menores que 125 μ s.



As rampas de subida e de descida de uma transição não devem exceder 30 V/ μ s. Taxas maiores do que esta podem induzir sinais em condutores adjacentes de um cabo.

Conversores de nível TTL – RS232

A maioria dos equipamentos digitais utilizam níveis TTL ou CMOS. Portanto, o primeiro passo para conectar um equipamento digital a uma interface RS232 é transformar níveis TTL (0 a 5 volts) em RS232 e vice-versa. Isto é feito por conversores de nível.

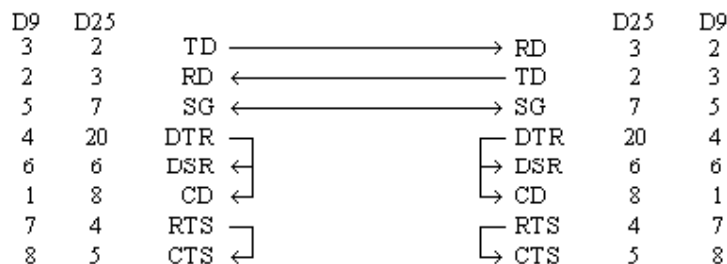
Existe uma variedade grande de equipamentos digitais que utilizam o driver 1488 (TTL => RS232) e o receiver 1489 (RS232 => TTL). Estes CIs contém 4 inversores de um mesmo tipo, sejam drivers ou receivers. O driver necessita duas fontes de alimentação +7,5 volts a +15 volts e -7,5 volts a -15 volts. Isto é um problema onde somente uma fonte de +5 volts é utilizada.

Um outro CI que está sendo largamente utilizado é o MAX232 (da Maxim). Ele inclui um circuito de “charge pump” capaz de gerar tensões de +10 volts e -10 volts a partir de uma fonte de alimentação simples de +5 volts, bastando para isso alguns capacitores externos, conforme pode-se observar na figura a seguir. Este CI também tem 2 receivers e 2 drivers no mesmo encapsulamento. Nos casos onde serão implementados somente as linhas de transmissão e de recepção de dados, não seria necessário 2 chips e fontes de alimentação extras.

Cabo “Null Modem”

Um cabo “null modem” é utilizado para conectar dois DTEs juntos. Isto é comumente usado como um meio barato para transferir arquivos entre computadores utilizando protocolos Zmodem, Xmodem, etc. Ele também pode ser utilizado em diversos sistemas de desenvolvimento.

Na figura abaixo é apresentado um método de conexão de um cabo “null modem”. Apenas 3 fios são necessários (TxD, RxD e GND). A teoria de operação é razoavelmente simples. O princípio é fazer o DTE pensar que está falando com um modem. Qualquer dado transmitido do DTE deve ser recebido no outro extremo e vice-versa. O sinal de terra (SG) também deve ser conectados ao terra comum dos dois DTEs.



O sinal DTR é conectado com os sinais DSR e CD nos dois extremos. Quando o sinal DTR for ativado (indicando que o canal de comunicação está aberto), imediatamente os sinais DSR e CD são ativados. Nessa hora o DTE pensa que o Modem Virtual ao qual está conectado está pronto e que foi

detectado uma portadora no outro modem. O DTE precisa se preocupar agora com os sinais RTS e CTS. Como os 2 DTEs se comunicam à mesma velocidade, o fluxo de controle não é necessário e consequentemente essas 2 linhas são conectadas juntas em cada DTE. Quando o computador quer transmitir um dado, ele ativa a linha RTS como estão conectadas juntas, imediatamente recebe a resposta que o outro DTE está pronto pela linha CTS.

Note que o sinal RI não está conectado em nenhum extremo. Esta linha é utilizada apenas para informar ao DTE que existe um sinal de chamada telefônica presente. Como não existe modem conectado a linha telefônica ela pode permanecer desconectada.

Controle do Fluxo de Dados

Se a conexão entre um DTE e um DCE for diversas vezes mais rápida do que a velocidade entre os DCEs, cedo ou tarde dados transmitidos do DTE serão perdidos, nesse caso o controle de fluxo de dados é utilizado. O controle de fluxo tem pode ser feito por hardware ou por software.

O controle do fluxo de dados por software, também conhecido como XON/XOFF utiliza 2 caracteres ASCII; XON representado pelo caracter ASCII 17 e XOFF representado pelo caracter ASCII 19. O modem tem normalmente um buffer pequeno e quando completado envia o caracter XOFF para avisar o DTE parar de enviar dados. Uma vez que o modem estiver pronto para receber mais dados ele envia o caracter XON e o DTE enviará mais dados. Este tipo de controle de fluxo tem a vantagem de não necessitar linhas adicionais, às linhas TxD e RxD. A desvantagem está no protocolo de comunicação que não poderá utilizar os caracteres ASCII 17 e 19 em suas mensagens.

O controle do fluxo de dados por hardware, também conhecido como RTS/CTS utiliza 2 linhas extras em seu cabo serial além das 2 linhas para transmissão de dados. Quando o DTE quer enviar dados ele ativa a linha RTS. Se o modem tem espaço para receber esse dado, ele irá responder ativando a linha CTS e o DTE começará a enviar dados. Se o modem não tem espaço para receber dados ele não ativa a linha CTS.

Edmur Canzian é engenheiro eletrônico formado pela Escola Politécnica da USP, professor dos cursos de extensão universitária sobre Microcontroladores da Família 8051, Microcontroladores PIC e Interfaces e Protocolos de Comunicação Serial na POLI-USP, professor do curso de pós-graduação em Automação Industrial na FEI/IECAT, e diretor da CNZ Engenharia e Informática Ltda.