

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**CLEBER SALVARO PERUCHI**

**COMUNICAÇÃO DE DADOS UTILIZANDO OS PADRÕES IrDA E  
COMUNICAÇÃO SERIAL RS232-C**

**CRICIÚMA, DEZEMBRO DE 2006.**

**CLEBER SALVARO PERUCHI**

**COMUNICAÇÃO DE DADOS UTILIZANDO OS PADRÕES IrDA E  
COMUNICAÇÃO SERIAL RS232-C**

Trabalho de conclusão de Curso apresentado para a obtenção do Grau de Bacharel em Ciência da Computação na Universidade do Extremo Sul Catarinense.

Professor Orientador: M.Eng. Evânio Ramos Nicoleit  
Professor Co-orientador: M.Sc. Paulo João Martins

**CRICIÚMA, DEZEMBRO DE 2006.**

*Dedico este trabalho a minha família, pois foi nela que encontrei a base de tudo o que me aconteceu. Dedico também aos meus falecidos avós que tinham um sonho de ter algum de seus netos formado em um curso superior. Espero que estejam vendo tudo o que está me acontecendo e juntos possamos comemorar felizes mais esta vitória.*

## **AGRADECIMENTOS**

### ***A Deus***

*Por tudo o que Ele tem feito por mim e pelas vezes que não soube entender o que as dificuldades significavam na minha vida. Sendo que vejo meu sonho realizado só tenho a dizer muito obrigado, Senhor, por ter me amado tanto a ponto de me dar uma profissão tão especial. Obrigado, por estar sempre ao meu lado nesta caminhada.*

### ***Aos meus pais***

*Pela educação que tive e pelos valores que me foram dados por meio do exemplo de sua vida ao longo do tempo. Sempre desejando minha felicidade e realização, tendo como base a simplicidade, amor, paciência e honestidade. Obrigado por tudo o que vocês fizeram por mim e ainda fazem, pelo apoio e pela lição de vida que me foi ensinada.*

### ***Aos meus amigos***

*Que acreditaram que eu era capaz de chegar aonde cheguei, que incentivaram e também questionaram se tudo o que estava me ocorrendo seria válido. Obrigado a todos que me apoiaram e motivaram a ir cada vez mais longe, nunca deixando que eu desanimasse.*

### ***A minha esposa***

*Você que perdeu noites de sono, sofria dia após dia com tudo que me acontecia, me ajudando a ter paciência e perseverança, nas coisas que eu fazia. Que Deus te proteja todos os dias e ilumine seus caminhos, guiando também ao seu objetivo maior. Agradeço por acreditar que eu fosse capaz, ainda que eu mesmo não acreditasse. Muito obrigado.*

*“... as pessoas podiam fechar os olhos diante da grandeza, do assustador da beleza, e podiam tapar os ouvidos diante da melodia ou de palavras sedutoras. Mas não podiam escapar ao aroma. Pois o aroma é um irmão da respiração. Com esta, ele penetra nas pessoas, elas não podem escapar-lhe caso queiram viver. E bem para dentro delas é que vai o aroma, diretamente para o coração, distinguindo lá categoricamente entre atração e menosprezo, nojo e prazer, amor e ódio. Quem dominasse os odores dominaria o coração das pessoas”.*

*Patrick Süskind*

**CLEBER SALVARO PERUCHI**

**COMUNICAÇÃO DE DADOS UTILIZANDO OS PADRÕES IrDA E  
COMUNICAÇÃO SERIAL RS232-C**

Submetido ao corpo docente do Departamento de Ciência da Computação da Universidade do Extremo Sul Catarinense como um dos requisitos para obtenção do Grau de Bacharel em Ciência da Computação.

---

Prof<sup>a</sup>. M.Sc. Ana Claudia Garcia Barbosa  
Coordenadora do Curso de Ciência da Computação

BANCA EXAMINADORA:

---

Prof. M.Eng. Evânio Ramos Nicoleit  
Orientador

---

Prof. M.Sc. Paulo João Martins  
Co-orientador

---

Prof<sup>a</sup>. M.Sc. Ana Claudia Garcia Barbosa  
1<sup>a</sup> Examinadora

---

Prof. Esp. Ênio José Peruchi  
2<sup>o</sup> Examinador

## RESUMO

Neste trabalho é apresentada a descrição de uma forma de comunicação entre computadores e entre outros dispositivos. Serão abordados os padrões de comunicação Serial (via RS232-C) e *Infrared Data Association* (IrDA), ambos mundialmente aceitos. Inicialmente é realizada uma revisão bibliográfica acerca de comunicação usando raios infravermelhos, porta IrDA e comunicação serial *Recommended Standard* (RS232-C). Posteriormente é apresentada uma interface de comunicação sem fio. Encontram-se também os procedimentos de desenvolvimento do dispositivo utilizado na comunicação proposta, bem como os esquemas relativos à construção do dispositivo de comunicação. Serão mostrados os tipos de comunicação utilizados, padrão RS232-C, comunicação serial e padrão IrDA.

**Palavras-chave:** IrDA, RS232-C, Comunicação sem fio.

## **ABSTRACT**

In this work is presented a communication way description among computers and even among other devices. The communication patterns approached are Serial Communication (RS232-C) and Infrared Data Association (IrDA), both globally accepted. Initially a revision is accomplished concerning communication using infrared rays, IrDA and serial communication Recommended Standard (RS232-C). Later a wireless communication interface is presented. They also are presented the procedures of development of the device used in the proposed communication, as well as the relative outlines the construction of the communication devices. They are presented also the communication types used, patterns RS232-C, serial communication and patterns IrDA.

**Key-words:** IrDA, RS232-C, Wireless Communication.

## LISTA DE FIGURAS

Figura 1. Diagrama de blocos de um sistema de transmissão .....	18
Figura 2. Modelo de Comunicação Simplificado.....	19
Figura 3. Exemplo de sinal analógico .....	21
Figura 4. Exemplo de sinal digital.....	22
Figura 5. Modo de operação <i>simplex</i> .....	22
Figura 5. Modo de operação <i>half-duplex</i> .....	22
Figura 7. Modo de operação <i>full-duplex</i> .....	23
Figura 8. Transmissão paralela.....	24
Figura 9. Transmissão serial.....	25
Figura 10. Transmissão serial assíncrona.....	26
Figura 11. Transmissão serial síncrona .....	28
Figura 12. Modelo conector RS232-C de 25 pinos (macho).....	30
Figura 13. Modelo conector RS232-C de 9 pinos (macho).....	30
Figura 14. Modelo do conector IrDA na placa mãe .....	38
Figura 15. Modelagem do dispositivo .....	40
Figura 16. Placa conversora com os componentes instalados 1 .....	42
Figura 17. Placa conversora com os componentes instalados 2.....	43
Figura 18. Diagrama de bloco TOIM4232.....	44
Figura 19. Foto TOIM4232 .....	45
Figura 20. Diagrama de bloco MAX3232E .....	45
Figura 21. Modelagem do dispositivo e seus componentes .....	46
Figura 22. Desenho da placa feito no <i>software ExpressPCB</i> .....	47
Figura 23. Placa com os componentes instalados .....	47
Figura 24. Placa IrDA com os componentes instalados .....	48

## LISTA DE TABELAS

Tabela 1. Características da transmissão sem fio, taxas e frequências.....	20
Tabela 2. Especificação técnica de cada pino RS232-C 25 pinos.....	30
Tabela 3. Especificação técnica de cada pino RS232-C 9 pinos.....	31
Tabela 4. Descrição da pinagem do conector serial DB 9 pinos do computador.....	31
Tabela 5. Taxa de dados IrDA.....	36
Tabela 6. Componentes necessários para montagem do dispositivo.....	41

## LISTA DE SIGLAS

ASCII	<i>American Standard Code for Information Interchange</i>
BPS	<i>Bits Per Second</i>
CD	<i>Carrier Detected</i>
CI	Circuito Integrado
CTS	<i>Clear To Send</i>
DCD	<i>Data Carrier Detected</i>
DCE	<i>Data Communication Equipment</i>
DDP	Diferença de Potencial
DSR	<i>Data Set Ready</i>
DTE	<i>Data Terminal Equipment</i>
DTR	<i>Data Terminal Ready</i>
EIA	<i>Electronic Industries Association</i>
ETD	<i>European Telework Development</i>
FIR	<i>Fast Infrared</i>
GND	<i>Ground</i>
IR	<i>Infrared</i>
IrDA	<i>Infrared Data Association</i>
MIR	<i>Medium Infrared</i>
QoS	<i>Quality of Service</i>
PC	<i>Personal Computer</i>
RD	<i>Received Data</i>
TD	<i>Transmitted Data</i>
RI	<i>Ring Indicator</i>
RS232-C	<i>Recommended Standart</i>
RTS	<i>Request To Send</i>
SIN	<i>Serial Input</i>

SIR	<i>Serial Infrared</i>
SOUT	<i>Serial Out</i>
TCC	Trabalho de Conclusão de Curso
TDM	<i>Time Division Multiplex</i>
TTL	<i>Transistor Transistor Logic</i>
VCC	<i>Volts Continuous Current</i>
VFIR	<i>Very Fast Infrared</i>

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>14</b>
1.1 OBJETIVOS.....	15
<b>1.1.1 Objetivo Geral .....</b>	<b>15</b>
<b>1.1.2 Objetivos Específicos.....</b>	<b>15</b>
1.2 JUSTIFICATIVA.....	15
1.3 ESTRUTURA DO TRABALHO .....	16
<b>2 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>18</b>
2.1 COMUNICAÇÃO DE DADOS.....	18
2.2 TIPOS DE SINAIS.....	19
<b>2.2.1 Sinais analógicos .....</b>	<b>20</b>
<b>2.2.2 Sinais digitais .....</b>	<b>21</b>
2.3 MODOS DE OPERAÇÃO.....	22
<b>2.3.1 Simplex.....</b>	<b>22</b>
<b>2.3.2 Half-duplex ou semi-duplex.....</b>	<b>22</b>
<b>2.3.3 Full-Duplex ou Duplex .....</b>	<b>23</b>
2.4 TIPOS DE TRANSMISSÃO .....	23
<b>2.4.1 Transmissão/comunicação paralela.....</b>	<b>23</b>
<b>2.4.2 Transmissão/comunicação serial.....</b>	<b>24</b>
2.4.2.1 Transmissão assíncrona.....	25
2.4.2.2 Transmissão síncrona .....	27
2.5 PADRÃO RS232-C.....	28
<b>2.5.1 Interface RS232-C .....</b>	<b>30</b>
<b>2.5.2 Descrição dos pinos da porta serial.....</b>	<b>31</b>
2.5.2.1 Sinais de entrada recebidos pelo computador .....	32
2.5.2.2 Sinais de saída enviados pelo computador .....	32
2.6 INFRAVERMELHO.....	33
<b>2.6.1 Contexto história do infravermelho.....</b>	<b>34</b>
<b>2.6.2 Características do infravermelho.....</b>	<b>35</b>
<b>2.6.3 IrDA .....</b>	<b>36</b>
<b>2.6.4 Barramento IrDA .....</b>	<b>37</b>

<b>2.6.5 Definição dos sinais .....</b>	<b>38</b>
<b>3 COMUNICAÇÃO DE DADOS UTILIZANDO OS PADRÕES IrDA E COMUNICAÇÃO SERIAL RS232-C .....</b>	<b>39</b>
3.1 METODOLOGIA .....	39
<b>3.1.1 Montagem.....</b>	<b>40</b>
<b>3.1.2 Descrição TOIM4232 .....</b>	<b>43</b>
<b>3.1.3 Descrição MAX3232E .....</b>	<b>45</b>
<b>3.1.4 Transceiver serial infravermelho .....</b>	<b>46</b>
<b>3.1.5 Funcionamento da placa .....</b>	<b>48</b>
3.2 CASO DA COMUNICAÇÃO NAS CALCULADORAS DAS FAMÍLIAS HP 48/49/50.....	49
<b>3.2.1 Protocolos de comunicação e transferência .....</b>	<b>50</b>
3.2.1.1 Protocolo de comunicação Kermit.....	50
3.2.1.2 Protocolo de comunicação Xmodem.....	52
3.3 RESULTADOS OBTIDOS.....	53
<b>4 CONCLUSÃO .....</b>	<b>55</b>
4.1 TRABALHOS FUTUROS.....	56
<b>REFERÊNCIAS .....</b>	<b>57</b>
<b>BIBLIOGRAFIA RECOMENDADA.....</b>	<b>58</b>
<b>ANEXOS .....</b>	<b>59</b>
<b>ANEXO A – MODELO DE APLICAÇÃO DO CIRCUITO UTILIZANDO PORTA RS232-C (ADAPTADOR EXTERNO INFRAVERMELHO) .....</b>	<b>60</b>
<b>ANEXO B – INDICAÇÃO DOS PINOS E DESCRIÇÃO TOIM4232 .....</b>	<b>61</b>
<b>ANEXO C – TAXA DE TRANSFERÊNCIA .....</b>	<b>62</b>

## 1 INTRODUÇÃO

A comunicação sempre foi essencial para a evolução da humanidade. Da palavra escrita até a chegada das telecomunicações, muitas formas de comunicação têm modificado a nossa sociedade. Conseqüentemente, mudaram também os procedimentos de nossas empresas.

Estando presente em muitos aspectos da nossa vida, pode ser encontrada em sistemas televisivos, quando nos comunicamos com outras pessoas, nas propagandas em *outdoors* entre outros. A todo o momento, algum tipo de comunicação está sendo estabelecido fazendo parte da natureza humana.

Há muito tempo pode-se comunicar com outras pessoas por meio de equipamentos que não necessitam de fios. Diante de toda essa evolução, e com esse intuito de avanços tecnológicos, pode-se dizer que a cada novidade tecnológica que surge, abrem-se novos caminhos e novos conceitos.

A comunicação de dados usando conexão por infravermelho surgiu a partir da busca por sistemas com bom desempenho e conexão confiável. Além destas características de transmissão, outra vantagem é o baixo custo para a sua implementação.

Atualmente as aplicações mais comuns nessa tecnologia se dão em comunicações entre câmeras digitais, impressoras, computadores desktops, notebooks, palmtops e periféricos em geral. A comunicação de dados pode ser feita nos dois sentidos da conexão.

O objetivo principal deste trabalho é realizar a comunicação entre dispositivos, mesmo que não possuam a interface de infravermelho, por meio da porta serial (RS232-C). Para isso foi desenvolvido um dispositivo de comunicação utilizando

os conceitos desta tecnologia, sendo possível estabelecer comunicação entre dois computadores ou periféricos em geral.

## 1.1 OBJETIVOS

Nesta sessão serão abordados os Objetivos Geral e Específico, Justificativa e Estrutura do Trabalho.

### 1.1.1 Objetivo Geral

Desenvolver um dispositivo para conexão entre periféricos usando o padrão IrDA por meio da porta serial, de forma a estabelecer a comunicação.

### 1.1.2 Objetivos Específicos

- a) compreender e documentar o funcionamento do infravermelho;
- b) conhecer o interfaceamento da porta serial e padrões de conversão de sinais;
- c) desenvolver um dispositivo capaz de estabelecer uma conexão para comunicação de dados IrDA/ RS232-C via porta serial, entre equipamentos;

## 1.2 JUSTIFICATIVA

Cada vez mais, é fundamental incorporar as inovações tecnológicas que o mundo das comunicações apresenta. Estas inovações geralmente estão associadas a conhecimentos prévios, buscando soluções nas mais diversas áreas (HELD, 1999). A

área em questão no projeto é a da comunicação entre dispositivos sem a necessidade de conexão física.

Considerando que nem todos os computadores e/ou periféricos possuem interfaces compatíveis (como o padrão IrDA e o padrão serial RS232-C), foi construído um dispositivo que seja capaz de fazer com que estes equipamentos possam se comunicar.

Foram efetuados testes entre computadores e periféricos como Calculadoras Gráficas HP famílias 48/49/50 e aparelho de Telefone Celular. Tipicamente estes equipamentos só têm conexão compatível entre si.

No caso do aparelho de Telefone Celular, este possuía interface de infravermelho acoplada. No entanto, o custo, por exemplo, de um cabo que conecta o celular ao computador, efetuando assim uma comunicação, fica em torno de R\$ 160,00 (Cento e Sessenta Reais) a R\$ 180,00 (Cento e Oitenta Reais), dependendo do modelo de cada aparelho, visto no *site* de um dos fabricantes de celular, a Nokia (<http://www.nokia.com.br> em 22/08/2006).

A busca por soluções para a compatibilidade da comunicação entre diferentes tipos de equipamentos é ao mesmo tempo a justificativa principal e o elemento motivador deste trabalho.

### 1.3 ESTRUTURA DO TRABALHO

No Capítulo 1 apresenta-se a Introdução juntamente com os objetivos geral e específico, a justificativa e estrutura do trabalho.

No Capítulo 2 do trabalho são apresentados fundamentos sobre comunicação de dados, tipos de sinais, modos de operação, tipos de transmissão, padrão RS232-C, infravermelho e IrDA.

No Capítulo 3 são descritos aspectos de implementação das soluções propostas bem como os protocolos de comunicação/transmissão utilizados. Também se encontram alguns resultados obtidos na comunicação entre computadores e periféricos utilizando esta forma de comunicação.

No Capítulo 4 são apresentadas e discutidas as conclusões e considerações finais acerca do trabalho desenvolvido e de trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Serão abordados neste capítulo conceitos acerca de comunicação, tipos de sinais, sincronismo, taxas de transferência, comunicação por infravermelho, padrão IrDA, comunicação serial e padrão RS232-C.

### 2.1 COMUNICAÇÃO DE DADOS

A comunicação de dados constitui o processo de transmissão de informação de um ponto a outro (ou mais pontos) por meio de sinais binários em fios, ou de ondas eletromagnéticas utilizando um código comum. Atualmente, vários aplicativos de multimídia que orientam a comunicação de dados em direção ao transporte de informações de voz, dados e vídeo vêm sendo cada vez mais pesquisados e desenvolvidos.

A comunicação é dependente de um meio de transmissão por onde os dados e informações possam ser enviados e recebidos, de acordo com a Figura 1.

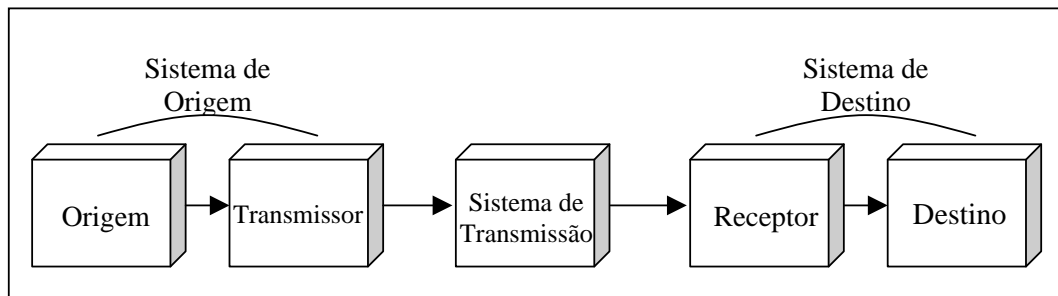


Figura 1. Diagrama de blocos de um sistema de transmissão  
Fonte: Stallings, W. (1997, p.5)

A comunicação pode ser efetivada de várias maneiras, isso dependerá do tipo de sinal que se está utilizando. Ela ocorre tanto por meios físicos (como utilização de linhas telefônicas por exemplo) ou por meio de ondas eletromagnéticas (sinais de

rádio frequência como ondas de rádio, infravermelho, dentre outros). Por exemplo, uma comunicação entre uma estação de trabalho e um servidor utilizando uma rede pública de telefonia, pode ser apresentado conforme visto na Figura 2.

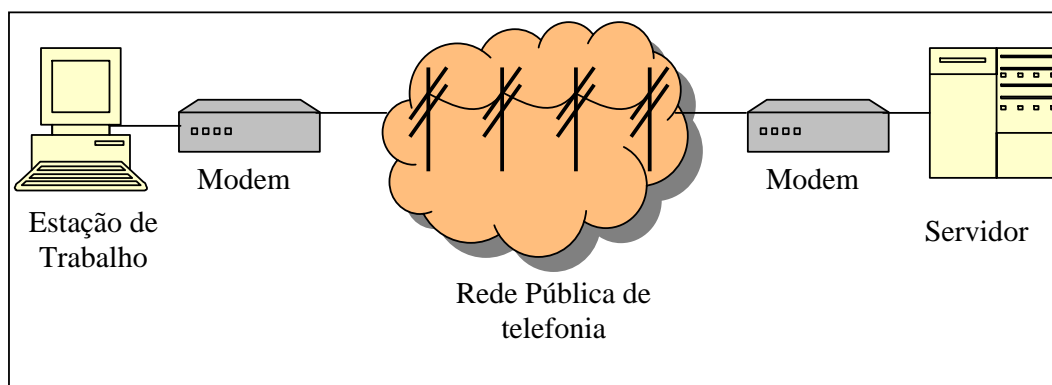


Figura 2. Modelo de Comunicação Simplificado  
Fonte: Stallings, W. (1997, p.5)

A comunicação de dados estuda os meios que podem ser utilizados para a transmissão destas mensagens digitais para outros dispositivos externos à fonte. Todo sistema de comunicação visa prover a maior taxa de transmissão (taxa de bits – dada em bits por segundo - *bps*) possível estabelecendo, desta forma, uma comunicação que possa ser estável e com a garantia de que a informação chegue ao destino intacta.

## 2.2 TIPOS DE SINAIS

Informações são um conjunto estruturado de dados visando alcançar o conhecimento. Dados são elementos de informação em forma apropriada para armazenamento, processamento e transmissão por meios automáticos sobre sinais elétricos. Os sinais podem ser classificados como sinais analógicos e sinais digitais.

A seguir é apresentada a Tabela 1 que contém as características de faixas de frequências de comunicação aplicadas. Nesta tabela apresentam-se as faixas de

freqüências utilizadas por cada tipo de forma de comunicação, sua denominação, tipo de modulação, tamanho da faixa e principais aplicações.

Tabela 1. Características da transmissão sem fio, taxas e freqüências

Faixas de Freqüência	Nome	Dados Analógicos		Dados Digitais		Principais Aplicações
		Modulação	Tamanho da Faixa	Modulação	Taxa de Dados	
30-300 kHz	LF ( <i>low frequency</i> )	Normalmente não utilizada		ASK, FSK, MSK	0.1 até 100 <i>bps</i>	Navegação
300-3000 kHz	MF ( <i>medium frequency</i> )	AM	Até 4 kHz	ASK, FSK, MSK	10 até 1000 <i>bps</i>	Comercial Rádio AM
3-30 MHz	HF ( <i>high frequency</i> )	AM, SSB	Até 4 kHz	ASK, FSK, MSK	10 até 3000 <i>bps</i>	Rádio de ondas curtas
30-300 MHz	VHF ( <i>very high frequency</i> )	AM, SSB, FM	5 kHz até 5 MHz	FSK, PSK	Até 100 <i>kbps</i>	Televisão VHF, Rádio FM
300-3000 MHz	UHF ( <i>ultra high frequency</i> )	FM, SSB	Até 20 MHz	PSK	Até 10 <i>Mbps</i>	Televisão UHF, microondas terrestre
3-30 GHz	SHF ( <i>super high frequency</i> )	FM	Até 500 MHz	PSK	Até 100 <i>Mbps</i>	Microondas terrestre, microondas de satélite
30-300 GHz	EHF ( <i>extremely high frequency</i> )	FM	Até 1 GHz	PSK	Até 750 <i>Mbps</i>	Ponto a ponto de curta distância

Fonte: Stallings, W. (1997, p.120)

### 2.2.1 Sinais analógicos

De acordo com Silveira (1991), os sinais elétricos assumem valores possíveis de amplitude, dependendo do meio de transmissão em que se encontram. Um exemplo de sinal analógico é a conversão de energia acústica, produzida pela voz, em energia elétrica num microfone. As vibrações sonoras que a voz produz na membrana da cápsula do microfone são transformadas em um sinal elétrico, variando

conforme o tempo e podendo assumir diversos valores, dependendo das vibrações recebidas por ele. Na Figura 3 encontra-se um exemplo de sinal analógico.

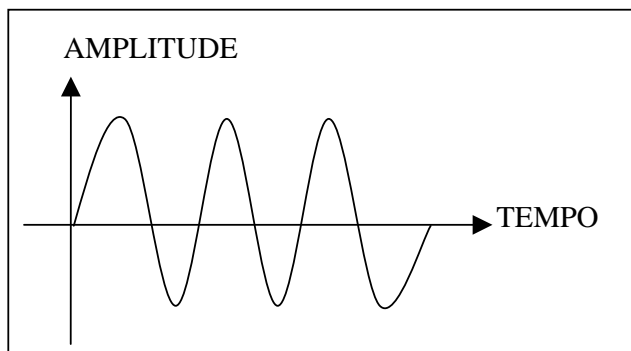


Figura 3. Exemplo de sinal analógico  
Fonte: Silveira, J. (1991, p.6)

### 2.2.2 Sinais digitais

Os sinais elétricos digitais, ao contrário dos analógicos, representam a informação assumindo valores de amplitude predeterminados em função do tempo (SILVEIRA, 1991). Sua utilização é normalmente encontrada em transmissão de dados. Na Figura 4 encontra-se um exemplo de sinal digital.

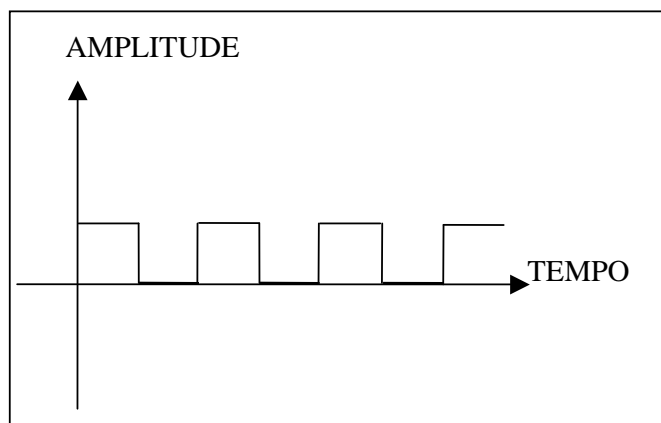


Figura 4. Exemplo de sinal digital  
Fonte: Silveira, J. (1991, p.7)

## 2.3 MODOS DE OPERAÇÃO

Nos sistemas de comunicações de dados, há a transmissão e a recepção, podendo elas acontecer de modo simultâneo ou não, recebendo as seguintes classificações: *Simplex*, *Half-Duplex* e *Full-Duplex*.

### 2.3.1 Simplex

Conforme Biondi (1996), esta comunicação ocorre de forma unidirecional, ou seja, em uma única direção possível, conforme mostra a Figura 5, não sendo possível a inversão do sentido de comunicação.

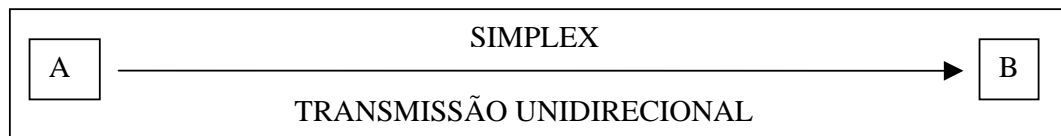


Figura 5. Modo de operação *simplex*  
Fonte: Silveira, J. (1991, p. 13)

### 2.3.2 Half-duplex

Ocorre de forma bidirecional sendo possível em ambas às direções, porém de modo não simultâneo entre os transmissores/receptores, como pode ser observada na Figura 6.



Figura 6. Modo de operação *half-duplex*  
Fonte: Silveira, J. (1991, p.13)

### 2.3.3 Full-Duplex ou Duplex

Neste tipo de comunicação a transmissão de dados ocorre de ambas as direções e de forma simultânea. É o mais poderoso dos três tipos apresentados anteriormente (BIONDI, 1996) conforme pode ser observado na Figura 7.

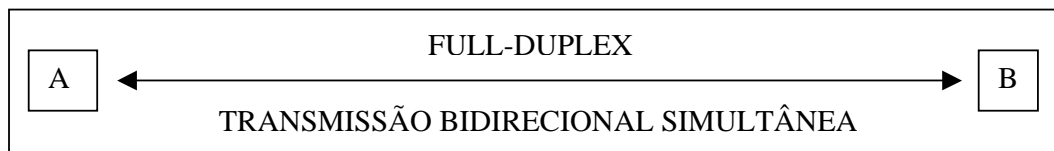


Figura 7. Modo de operação *full-duplex*  
Fonte: Silveira, J. (1991, p.13)

## 2.4 TIPOS DE TRANSMISSÃO

A transmissão de dados entre equipamentos pode ser feita de variadas formas. Dentre elas podem ser citadas a transmissão paralela e a serial. A seguir serão descritas as características da transmissão paralela, serial, serial assíncrona e serial síncrona.

### 2.4.1 Transmissão/comunicação paralela

De acordo com Silveira (1991), é o tipo de transmissão onde vários *bits* são transmitidos simultaneamente, conforme Figura 8. Este método é muito utilizado em conexões entre computadores e periféricos próximos. Em casos onde as distâncias são maiores, este tipo de comunicação torna-se inadequada. Quanto maior for a distância dos cabos, mais freqüentes serão as interferências e ruídos associados. Segundo Biondi (1996), uma distância de segurança pode ser fixada entre 3 a 20 metros, mas isso dependerá do dispositivo que estiver sendo conectado.

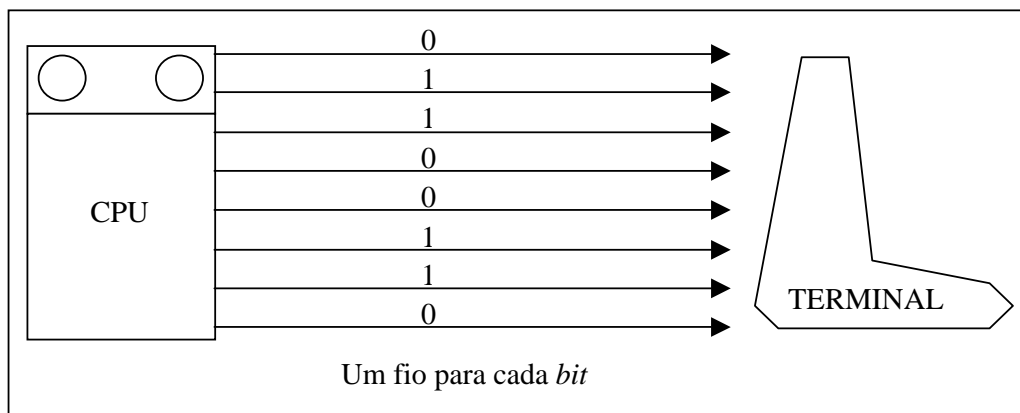


Figura 8. Transmissão paralela  
 Fonte: Silveira, J. (1991, p.14)

Na comunicação paralela é enviada uma palavra de dados por vez ao longo de um barramento. A comunicação com impressoras é um exemplo típico de uma comunicação paralela, sendo que a velocidade é alta e as distâncias são curtas. Juntamente com o envio dos dados, deve-se também enviar sinais de controle e de estado para que se faça a verificação de que os dados estão chegando ao seu destino. As informações são transmitidas em várias filas de dados.

#### 2.4.2 Transmissão/comunicação serial

A comunicação é feita por meio de uma única linha de dados, podendo ser feito tanto transmissão quanto recepção. A velocidade deste tipo de comunicação é bem inferior à comunicação paralela. Um exemplo que pode ser citado é o de um modem, onde o tipo de comunicação associado é de forma paralela. A taxa de transferência é medida em *bits/segundo* (*bps* ou *baud rate*), sendo que esta velocidade pode variar entre 100 *bps* e 115.2 *Kbps* em comunicações seriais típicas.

Os canais de comunicação são meios utilizados para enviar o sinal de um emissor (fonte) a um receptor. O suporte pode ser um meio físico (um fio, conectando

dispositivos de comunicação, por exemplo), ou por ondas eletromagnéticas (ondas de rádio, infravermelho, *laser* ou outra).

Neste tipo de transmissão os *bits* são transmitidos de maneira seqüencial, ou seja, um após o outro até que se formem as palavras de dados, conforme apresentado na Figura 9.

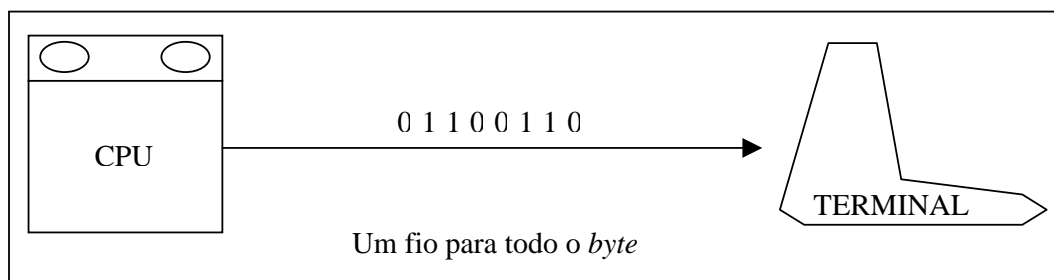


Figura 9. Transmissão serial  
Fonte: Silveira, J. (1991, p.14)

Conforme Zelenovsky e Mendonça (1999), a porta COM1 do PC está mapeada no endereço 3F8h, a porta COM2 no endereço 2F8h, a porta COM3 no endereço 3E8h e a porta COM4 no endereço 2E8h. Os dados são transmitidos pelo circuito, em seqüência, do *bit* menos significativo para o mais significativo.

A porta serial pode ser conectada com apenas dois condutores em uma conexão unidirecional ou com três condutores em uma conexão bidirecional.

#### 2.4.2.1 Transmissão assíncrona

Comunicação serial assíncrona é a forma mais utilizada na transmissão de dados, não necessitando de sincronização entre os dispositivos, uma vez que seus caracteres serão transmitidos individualmente e não em blocos.

Os computadores enviam e recebem as informações intercaladas por *bits* especiais. Em toda a comunicação que se faz, existem sinalizadores que indicam o início de um caractere chamado de *Start bit* (sempre 0 = zero) e um outro para indicar o fim do

caractere chamado de *Stop bit* (sempre 1 = um), estão localizados no início e no final de cada caractere respectivamente. Conforme Silveira (1991), o *Start*, que é o *bit* de partida, corresponde a uma interrupção de sinal na linha e o *Stop*, que é o *bit* de parada, indica a condição de marca ou repouso, neste caso, a existência do sinal na linha.

Normalmente o *Stop* possui tempo equivalente a 1,4 ou 2,0 vezes o tempo de *Start*, conforme ilustrado na Figura 10. Podem ocorrer casos em que se tenha até dois *Stop* no final. Existe também um *bit* especial chamado paridade, onde seus dados mais comuns são: nenhuma (*none*), ímpar (*even*) ou par (*odd*), tendo como principal finalidade detectar os erros de transmissão. É importante saber que os *bits* são transmitidos em série, neste caso, um após o outro.

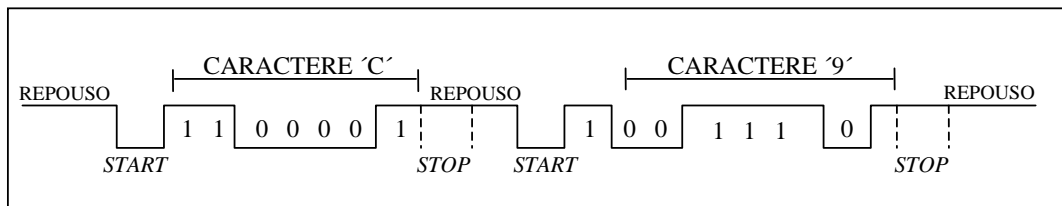


Figura 10. Transmissão serial assíncrona  
Fonte: Silveira, J. (1991, p.15)

O termo assíncrono, neste caso, refere-se ao fato de haver irregularidade entre os instantes de ocorrência dos caracteres, isso quer dizer que o aparelho que envia estes dados pode variar o tempo de emissão entre cada caractere sem que o aparelho receptor saiba. O que realmente acontece dentro deste tipo de transmissão é que o receptor só saberá da existência da comunicação quando receber o *bit Start* e por meio de seu *clock*, ele irá sincronizar seus circuitos elétricos, lendo assim, cada *bit* no seu momento apropriado, de acordo com Biondi (1996), pode acontecer em certos momentos a possibilidade de nenhuma informação estar sendo transmitida, como no exemplo a seguir:

Mensagem: “OI”

Código ASCII do caractere “O”: 79 decimal = 1001111 binário.

Código ASCII do caractere “I”: 73 decimal = 1001001 binário.

Formato da mensagem a ser transmitida:

Ⓢ 1111001 Ⓢ Ⓢ 1001001 Ⓢ

Apesar de ser assíncrono o ritmo da transmissão e, a emissão dos caracteres ser irregular, mesmo assim ele possui um sincronismo ao nível dos *bits* que formam o caractere, isso se dá ao fato da identificação do *Start*, por meio do receptor, conhecendo assim, os instantes que separam os *bits* dentro dos caracteres. Segundo Silveira (1991), levando em consideração os possíveis erros que esta comunicação possui, ela é utilizada normalmente em transmissões de dados em que as taxas de transmissão ficam abaixo de 2.4 *Kbps*. Podendo, no entanto ter taxas maiores, como por exemplo, velocidades de até 115.2 *Kbps*, mas a comunicação ficaria instável devido aos ruídos existentes e a distância estabelecida entre os equipamentos. Por este motivo o custo destes aparelhos é bem menor que os aparelhos síncronos, levando em consideração também serem de fácil fabricação.

#### 2.4.2.2 Transmissão síncrona

Na transmissão sincronizada não existe a presença dos caracteres *Start-Stop* e nem tempo de repouso entre eles, pois os dados são enviados um após o outro imediatamente, esta comunicação é estabelecida por meio de um ritmo fixo na transmissão dos *bits* de todo um bloco de caracteres, permitindo assim um ganho substancial nas comunicações. Nesta comunicação o transmissor e o receptor devem ser sincronizados para haja a troca de comunicação de dados. Normalmente é utilizada uma palavra de sincronismo para que ambos os dispositivos ajustem seus relógios internos.

Silveira (1991), cita que depois que a sincronização é feita, os *bits* são enviados seqüencialmente, até uma quantidade pré-estabelecida entre os dispositivos. Se enviarmos 1.204 caracteres sincronamente, serão gastos somente 8.192 *bits*, enquanto nos equipamentos assíncronos (com *Start* e *Stop bits*) serão necessários 10.240 *bits*. Neste exemplo o ganho chega a 20% nas comunicações, sendo que os dados são enviados de forma contínua e preparados em blocos, podendo ter diferentes tamanhos.

Ao iniciar uma transmissão o equipamento transmissor envia uma configuração de *bits* de sincronização, Figura 11, colocando o aparelho receptor em sincronia com o mesmo, sendo esta configuração de *bits* de sincronização diferente de qualquer outro bloco da mensagem a ser enviada.

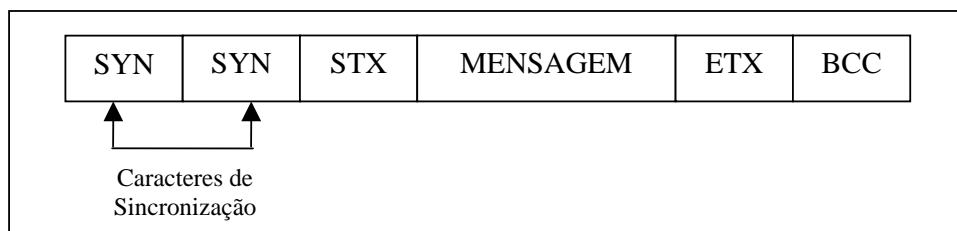


Figura 11. Transmissão serial síncrona  
Fonte: Silveira, J. (1991, p.16)

Por se tratar de uma transmissão padronizada, há a necessidade de utilizar protocolos que garantam o sincronismo destes equipamentos, tornando o custo destes mais elevados, pois eles devem possuir *buffers* de armazenamento para garantirem que a velocidade seja constante, tendo pulsos de mesma duração. Estes aparelhos normalmente possuem técnicas mais aperfeiçoadas de detecção de erros, permitindo passar ainda mais informações por período de tempo.

## 2.5 PADRÃO RS232-C

Este padrão é muito antigo, segundo Zelenovsky e Mendonça (1999), e também muito popular, usado para conectar dois equipamentos de comunicação de

dados como, por exemplo: terminais, *plotters*, osciloscópios, unidades de fita, impressoras entre outros. Sendo que para uma aplicação prática qualquer, será necessário utilizar alguns circuitos que possam converter os níveis elétricos TTL (*Transistor Transistor Logic*) para os necessariamente exigidos pela interface, convertendo também o caminho inverso.

A comunicação de dados é feita de forma serial, ou seja, os dados são transmitidos por uma única via um após o outro. A primeira especificação RS232 definia o comprimento máximo do cabo serial de até 15 metros a uma velocidade máxima de 20 *Kbps*, na época da definição do padrão, 20 *Kbps* era uma velocidade alta para este tipo de comunicação. A especificação evoluiu com a revisão C (RS232-C) e permitiu a utilização de velocidades até 115 *Kbps* a distâncias de até 60 metros.

A Associação das Indústrias Eletrônicas (*Electronic Industries Association – EIA*), dos Estados Unidos, publicou um padrão sobre a interface serial a ser utilizada para interconectar *European Telework Development* (ETD) e modem.

Esse padrão recebeu o nome de RS232 e às vezes é referenciado como EIA-RS232-C (revisão C) ou EIA-RS-232-D (revisão D, de 1987).

Ele define as características elétricas dos sinais, características mecânicas bem como a função de cada sinal.

Neste tipo de padrão o caminho é unidirecional, ponto a ponto, para uma distância de até 15,24 m, e possui uma taxa de máxima de transmissão de até 20 *Kbits/s* (ZELENOVSKY; MENDONÇA, 1999). Usando um exemplo simples de conversão tem-se o seguinte: nível alto TTL é convertido em -12 V e nível baixo TTL em +12 V, onde o TTL significa o tempo em que este processo ficará vivo ou ativo na rede até que ele seja retirado, prevenindo o congestionamento da mesma.

### 2.5.1 Interface RS232-C

Os valores lógicos dependem da quantidade de tensão recebida. Considera-se que os computadores e terminais que emitem e recebem informação são *Data Terminal Equipment* (DTE), e que os equipamentos de transmissão de informação são *Data Communication Equipment* (DCE). Na Figura 12 tem-se um modelo do conector de 25 pinos macho de uma porta serial.

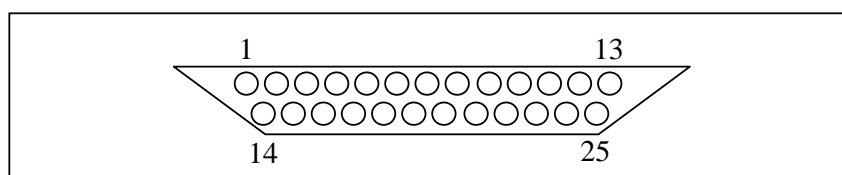


Figura 12. Modelo conector RS232-C de 25 pinos (macho)  
Fonte: <http://www.cnz.com.br>

A seguir na Tabela 2, apresenta-se a relação das especificações técnicas da pinagem deste conector com suas respectivas funções.

Tabela 2. Especificação técnica de cada pino RS232-C de 25 pinos

1 - <i>Ground</i> de proteção	9/10 – Reservados	18 – Desligado
2 – TxD	11 – Desligado	19 - RTS secundário
3 – RxD	12 – DCD secundário	20 – <i>Data terminal ready</i> DTR
4 - <i>Request to send</i> RTS	13 – CTS secundário	21 - <i>Signal quality detector</i>
5 - <i>Clear to send</i> CTS	14 – TxD secundário	22 - <i>Ring indicator</i>
6 – <i>Data set ready</i> DSR	15 - Tx <i>signal timing</i>	23 – <i>Data signal rate select</i>
7 – <i>Ground</i>	16 – RxD secundário	24 - Tx <i>signal timing</i>
8 – <i>Data carrier detected</i> DCD	17 - Rx <i>signal timing</i>	25 – Desligado

Fonte: <http://www.cnz.com.br>

Na Figura 13 é mostrado um modelo do conector de 9 pinos macho de uma porta serial.

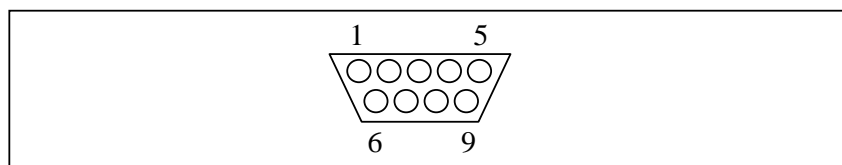


Figura 13. Modelo conector RS232-C de 9 pinos (macho)  
Fonte: <http://www.cnz.com.br>

É apresentado na Tabela 3, as especificações técnicas da pinagem deste conector com suas respectivas funções.

Tabela 3. Especificação técnica de cada pino RS232-C de 9 pinos

1 – <i>Data carrier detected</i> DCD	6 - <i>Data set ready</i> DSR
2 – RxD	7 - <i>Request to send</i> RTS
3 – TxD	8 - <i>Clear to send</i> CTS
4 – <i>Data terminal ready</i> DTR	9 - <i>Ring indicator</i> RI
5 – <i>Ground</i>	

Fonte: <http://www.cnz.com.br>

### 2.5.2 Descrição dos pinos da porta serial

Os sinais da porta serial do computador estão disponíveis tipicamente em um conector tipo DB macho de 9 pinos. Conforme mostra a Tabela 4, pode-se ver a seqüência de pinos com suas respectivas funções.

Tabela 4. Descrição da pinagem do conector serial DB 9 pinos do computador

9 Pinos	Sinal	PC
1	CD- <i>Carrier Detect</i>	←
2	RD- <i>Received Data</i>	←
3	TD- <i>Transmitted Data</i>	→
4	DTR- <i>Data Terminal Ready</i>	→
5	GND- <i>Ground</i>	
6	DSR- <i>Data Set Ready</i>	←
7	RTS- <i>Request to Send</i>	→
8	CTS- <i>Clear to Send</i>	←
9	RI- <i>Ring Indicator</i>	←

Fonte: Zelenovsky, Mendonça (1999, p.459)

### 2.5.2.1 Sinais de entrada recebidos pelo computador

São aqueles sinais gerados por outro dispositivo serial e recebidos pelo computador, depois de passarem por um conversor RS232-C são entregues ao circuito (ZELENOVSKY; MENDONÇA, 1999). Mostra-se a seguir a descrição da finalidade de cada um deles:

RD – *Received Data*: é por aonde chegam os dados seriais (SIN – *Serial Input*).

CTS – *Clear To Send*: é por meio desta linha que o periférico indica seu estado de pronto para transmitir os dados. É possível gerar uma interrupção quando o CTS muda de estado.

DSR – *Data Set Ready*: por meio desta linha o periférico indica seu estado de pronto para se comunicar. É possível gerar uma interrupção quando a entrada DSR muda de estado.

CD – *Carrier Detect*: por meio desta linha o periférico informa que detectou a portadora. É possível gerar uma interrupção quando a entrada CD muda de estado.

RI – *Ring Indicator*: neste caso o periférico informa o recebimento do tom de discar de uma linha telefônica. É possível gerar uma interrupção com sua transição, de alto para baixo.

### 2.5.2.2 Sinais de saída enviados pelo computador

Conforme Zelenovsky e Mendonça (1999), são aqueles sinais gerados pelo computador e enviados para um outro dispositivo serial, onde os dados são gerados no circuito, depois de passar por um conversor RS232-C e entregues

posteriormente ao conector serial. Apresenta-se a seguir uma descrição da finalidade de cada um deles:

TD – *Transmitted Data*: é por onde são enviados os dados seriais (SOUT – *Serial Out*), sendo sua saída SOUT alterada para o estado de marca (-12 V) após a inicialização do mesmo (*reset*).

RTS – *Request To Send*: indica para o periférico que o circuito está pronto para transmitir os dados, sua saída RTS é colocada em alto (+12 V = ativa) após a operação de inicialização (*reset*).

DTR – *Data Terminal Ready*: indica para o periférico que o circuito está pronto para se comunicar, sua saída DTR é colocada em alto (+12 V = ativa) após a operação de inicialização (*reset*).

## 2.6 INFRAVERMELHO

Utilizado há alguns anos para comunicação e controle de equipamentos comandados à distância, vem evoluindo muito nas tecnologias de controle remoto e de sinais. Podem ser encontrados no mercado em celulares, câmeras digitais, *mouses*, teclados, impressoras entre outros dispositivos, podendo ainda ser controlados pelo computador.

Isso tudo para oferecer mais vantagens em relação às tecnologias que ainda utilizam fios em sua comunicação, sendo que desta forma, o usuário limita-se apenas a distância alcançada por este dispositivo.

Segundo Knutson e Brown (2004), a especificação de infravermelho utiliza comprimentos de onda entre 750 e 1000 nanômetros. O infravermelho foi projetado para ser usado em áreas fechadas e opera com transmissões não

direcionadas que alcançam um máximo de aproximadamente 10 metros, caso não exista luz do sol interferindo, ou 20 metros, caso sejam utilizados receptores mais sensíveis.

Seu custo é um pouco mais alto do que as tecnologias que utilizam fios, devido aos componentes e a fonte geradora de energia para o funcionamento do mesmo, cabendo a cada usuário avaliar se este valor torna-se ou não, inviável a sua aquisição, levando em consideração a relação custo-benefício.

### **2.6.1 Contexto histórico do infravermelho**

Sir William Herschel, um astrônomo muito conceituado, descobriu o infravermelho em 1800. Construiu seus próprios telescópios ficando, portanto, bastante familiarizado com lentes e espelhos (KNUTSON; BROWN, 2004). A história da comunicação de dados por infravermelho na verdade começa a mais de 150 anos antes do primeiro controle remoto de televisão por infravermelho, e quase 200 anos antes do IrDA.

Sabendo que a luz solar continha todas as cores do espectro e que era também uma fonte de calor, Herschel queria descobrir qual ou quais as cores responsáveis pelo aquecimento dos objetos. Idealizou um experimento usando um prisma, papelão e termômetros com bulbos pretos onde mediu as temperaturas das diferentes cores. Herschel observou um aumento de temperatura à medida que movia o termômetro de violeta para o vermelho no espectro criado pela luz do sol atravessando o prisma. Descobriu que a temperatura mais quente ocorria, de fato, além da luz vermelha.

A radiação que causou esse aquecimento não era visível; Herschel denominou essa radiação invisível como “raios caloríficos”. Hoje, é conhecida como infravermelha.

## 2.6.2 Características do infravermelho

O infravermelho tem como característica bom desempenho em termos de taxa de bits, porém um curto alcance. Esta tecnologia necessita de um caminho livre para poder estabelecer a comunicação de dados ponto a ponto.

As aplicações mais comuns nessa tecnologia nos dias de hoje se dão em ligações entre câmeras digitais, impressoras, computadores desktops, notebooks, palmtops, periféricos em geral. A comunicação de dados pode ser feita nos dois sentidos, máquina-periférico, periférico-máquina.

Neste tipo de tecnologia as transmissões de dados são feitas em frequências muito altas, sendo um pouco abaixo da luz visível, não podendo penetrar em objetos que sejam opacos, tornando-se assim transmissões difusas ou diretas. Conforme Biondi (1996), o infravermelho possui comprimento de onda entre 1 mm e 1 $\mu$ m, ele está associado ao calor, sendo que são muitas vezes usados para rastreamento durante a noite.

Conforme Knutson e Brown (2004), as características do IrDA podem ser divididas em três partes: frequência, tempo e espaço.

Frequência: enquanto as transmissões são feitas separadamente, não coincidindo as frequências, não ocorrerão conflitos. Caso os dispositivos transmitam na mesma frequência, desta maneira ter-se-á interferência.

Tempo: um dos dispositivos deve estar transmitindo um sinal e o outro recebendo, dentro da mesma fração de tempo, para que haja a comunicação. No caso de haver mais de um dispositivo no mesmo local desejando comunicar-se com outro equipamento, deverá ser aplicada uma divisão de tempo chamada de Multiplexação por Divisão de Tempo (TDM), controlando quando cada dispositivo deverá transmitir.

Espaço: envolve uma forma de separação do espaço para que as frequências não interfiram umas nas outras, podendo ser repetidas em outro local diferente. Isso faz com que os sinais sejam isolados de forma a permitir estabilidade na comunicação.

Além do infravermelho, existem outras tecnologias que utilizam a comunicação sem fio como é o caso das ondas de rádio, frequência de ondas recebidas pelos televisores, comunicação com satélites e outras mais. Para este projeto optou-se pela tecnologia do infravermelho pela sua praticidade e ampla abertura de conhecimento no meio tecnológico.

Seu alcance está limitado a distâncias pequenas de aproximadamente 1,5 metros.

As taxas de transmissão de dados variam conforme o método utilizado, podendo ser definidos os seguintes valores: 2,4 *Kbps*, 9,6 *Kbps*, 19,2 *Kbps*, 38,4 *Kbps*, 57,6 *Kbps*, 115,2 *Kbps*, chamadas de *Serial Infrared (SIR)*. Existem também as chamadas *Fast Infrared (FIR)*, *Medium Infrared (MIR)* e *Very Fast Infrared (VFIR)*, variando nestas a velocidade de comunicação conforme Tabela 5. Estas taxas são utilizadas para pequenas transmissões de dados conforme cita Knutson e Brown (2004).

Tabela 5. Taxa de dados IrDA

	TAXA DE DADOS
SIR	2,4 – 9,6 – 19,2 – 38,4 – 57,6 – 115,2 <i>Kbps</i>
MIR	0,576 – 1.152 <i>Mbps</i>
FIR	4 <i>Mbps</i>
VFIR	16 <i>Mbps</i>

Fonte: Knutson e Brown (2004, p. 23)

### 2.6.3 IrDA

*Infrared Data Association (IrDA)* é uma organização que define padrões de comunicação entre equipamentos, comunicação wireless permitindo a comunicação sem

firos, por meio de radiação infravermelha. O padrão IrDA estabelece uma interface de comunicação entre equipamentos eletrônicos, sendo útil no controle de periféricos e na transferência de dados. Permite a conexão de dispositivos sem fio ao computador (ou equipamento com tecnologia apropriada), tais como impressoras, telefones celulares, notebooks e PDAs. As transmissões podem ser do tipo *full-duplex* ou *half-duplex*.

Os padrões associados à esta tecnologia são:

- 1.0 - com taxas de transmissão de até 115.200 *bps*;
- 1.1 - com taxas de transmissão de até 4.194.304 *bps* (4 *Mbps*).

#### **2.6.4 Barramento IrDA**

Nas placas-mãe existentes no mercado, de acordo com um site de eletrônica muito conceituado, [www.farnell.com.br](http://www.farnell.com.br), pode ser observado que algumas possuem um conector na própria placa para uso do barramento IrDA, conforme esquema apresentado na Figura 14. Isto permite a comunicação sem fio entre o micro e os periféricos que possuam sensores infravermelhos, podendo se conectar a alguns modelos de impressoras, *notebooks*, entre outros. Porém, nem todos os fabricantes de placas-mãe do mercado introduzem este dispositivo em sua fabricação. É necessário, portanto, verificar o *lay-out* da mesma e se este tipo de periférico está disponível no modelo da placa. As fabricantes ASUS e SOYO utilizaram este tipo de conexão anos atrás. Atualmente, somente alguns raros modelos comportam esta tecnologia.

No intuito de utilizar este barramento, será necessário instalar um adaptador no conector IrDA que existe na placa-mãe, sendo que este adaptador não vem junto com a mesma.

Pin	Signal Name
1	+5 V
2	no connect
3	IRRX
4	Ground
5	IRTX

Figura 14. Modelo do conector IrDA na placa mãe  
Fonte: <http://www.farnell.com.br>

### 2.6.5 Definição dos sinais

A cada pino do conector de interface corresponde um circuito cuja denominação seguida é aquela da recomendação CCITT V24.

Os circuitos são divididos em quatro categorias:

- a) circuitos de dados;
- b) circuitos de sincronismo;
- c) circuitos de controle;
- d) circuitos de teste.

Com base nestes conhecimentos, tem-se um embasamento mais amplo de como efetuar este tipo de conexão, proporcionando desta maneira a comunicação entre os dispositivos. A seguir será apresentado a implementação e desenvolvimento do dispositivo de comunicação, bem como, exemplos de comunicação dos periféricos e seus protocolos utilizados.

### **3 COMUNICAÇÃO DE DADOS UTILIZANDO OS PADRÕES IrDA E COMUNICAÇÃO SERIAL RS232-C**

Tendo como referência à compatibilidade de comunicação por meio de infravermelho com uma grande quantidade de periféricos encontrados hoje no mercado, seu baixo custo em função dos componentes serem comuns de estarem a disposição, foi utilizado este tipo de interfaceamento.

Como toda regra tem sua exceção, tem também a desvantagem deste tipo de periférico. Sua segurança é volátil, deixando a desejar. E a distância se torna um ponto que deve ser respeitado e analisado. Ou seja, ao longo da comunicação pode haver momentaneamente desconexões, pois se trata de um dispositivo de transmissão de dados que necessita de visada direta.

#### **3.1 METODOLOGIA**

A metodologia que foi utilizada para o desenvolvimento do sistema seguiu algumas etapas. O objetivo foi o de buscar a adequação no desenvolvimento desse trabalho com os procedimentos necessários à execução de um sistema em hardware para estabelecer a conexão entre uma porta RS232 e uma unidade IrDA (ANEXO A).

Em seguida destaca-se cada uma delas, estando todas em conformidade com a seqüência mostrada a seguir:

- a) revisão bibliográfica acerca dos assuntos envolvidos no trabalho em Artigos, Revista Técnicas, TCCs, Monografias e Internet;
- b) implementação de dispositivos de interfaceamento RS232/IrDA, IrDA/RS232 envolvidos no processo.

- c) realização de testes envolvendo os padrões de comunicação utilizados;
- d) análise de uma aplicação.

### 3.1.1 Montagem

O elemento central do projeto é o CI TOIM4232. Trata-se de um *encoder/decoder* que converte o sinal do padrão RS232 para IrDA e vice-versa na transmissão e na recepção. O IrDA baseado no CI TOIM4232 pode ser programado para operar em taxas de bits 1200 bit/s a 115.2 kbit/s pelo software de comunicação via porta RS232. Optou-se em deixá-lo operando na faixa de 9600 bit/s. O CI U3 para uma primeira aplicação não foi utilizado, sendo substituído por um circuito com componentes discretos. O TOIM4232 necessita de um gerador de *clock* externo com frequência de 3.6864 MHz. O diagrama esquemático do dispositivo de interface é apresentado na Figura 15.

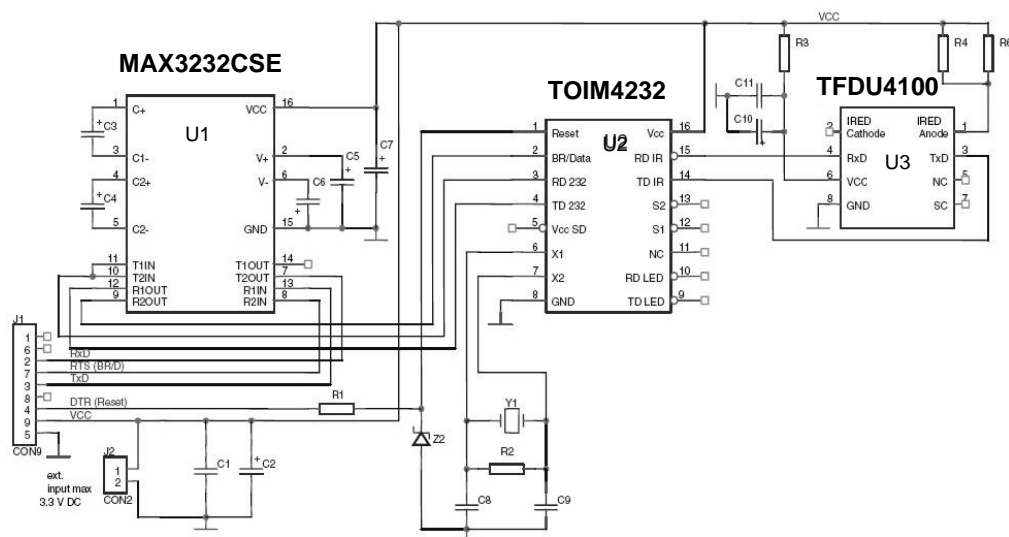


Figura 15. Modelagem do dispositivo  
Fonte: www.vishay.com

Para isso, foram adquiridos os componentes que estão relacionados na Tabela 6, conforme a necessidade do protótipo.

Tabela 6. Componentes necessários para montagem do dispositivo

	Componentes	Valores Recomendados
1	C1	100 nF
2	C2	10 μF, 16 V
3	C3	100 nF
4	C4	100 nF
5	C5	100 nF
6	C6	100 nF
7	C7	1 μF, 16 V
8	C8	22 pF
9	C9	22 pF
10	C10	6,8 μF, 16 V
11	C11	100 nF
12	Z2	3,6 V
13	R1	5,6 Ω
14	R2	100 kΩ
15	R3	47 Ω
16	R4	20 Ω
17	R6	20 Ω
18	Y1	3,686400 MHz
19	U1	MAXIM MAX 3232E
20	U2	TOIM 4232
21	U3	-----
22	J1	Conector 9 Pinos
23	J2	Conector de energia (Alimentação)

Fonte: [www.vishay.com](http://www.vishay.com)

Onde os componentes iniciados com a letra “C” são os capacitores, os iniciados com a letra “Z” é o diodo zener, com a letra “R” são os resistores, o “Y1” um cristal de frequência, o “U1” é o componente MAX3232E, o “U2” o componente TOIM4232, o “U3” está associado à placa do *transceiver* serial infravermelho, o “J1” é o conector de 9 pinos utilizado para conectar a placa na porta serial do computador. Foram construídas duas placas iguais, uma para cada computador/periférico, onde as mesmas serão conectadas em suas saídas seriais (as mesmas do *mouse* serial). Na Figura 16 tem-se a primeira placa montada em um *protoboard*.

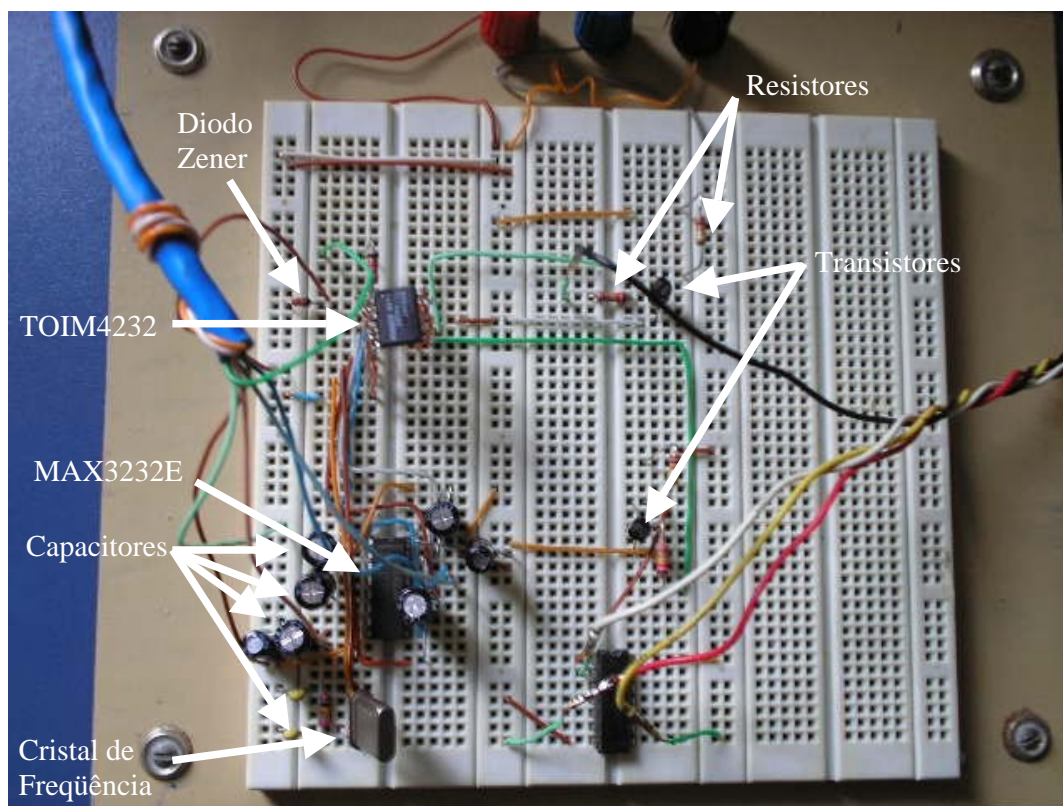


Figura 16. Placa conversora com os componentes instalados 1

Na segunda placa estão dispostos os mesmos componentes da primeira placa de acordo com a Figura 17, onde esta poderá se comunicar com a outra placa por meio de raios infravermelhos emitidos pelo transmissor e recebidos no destino pelo receptor.

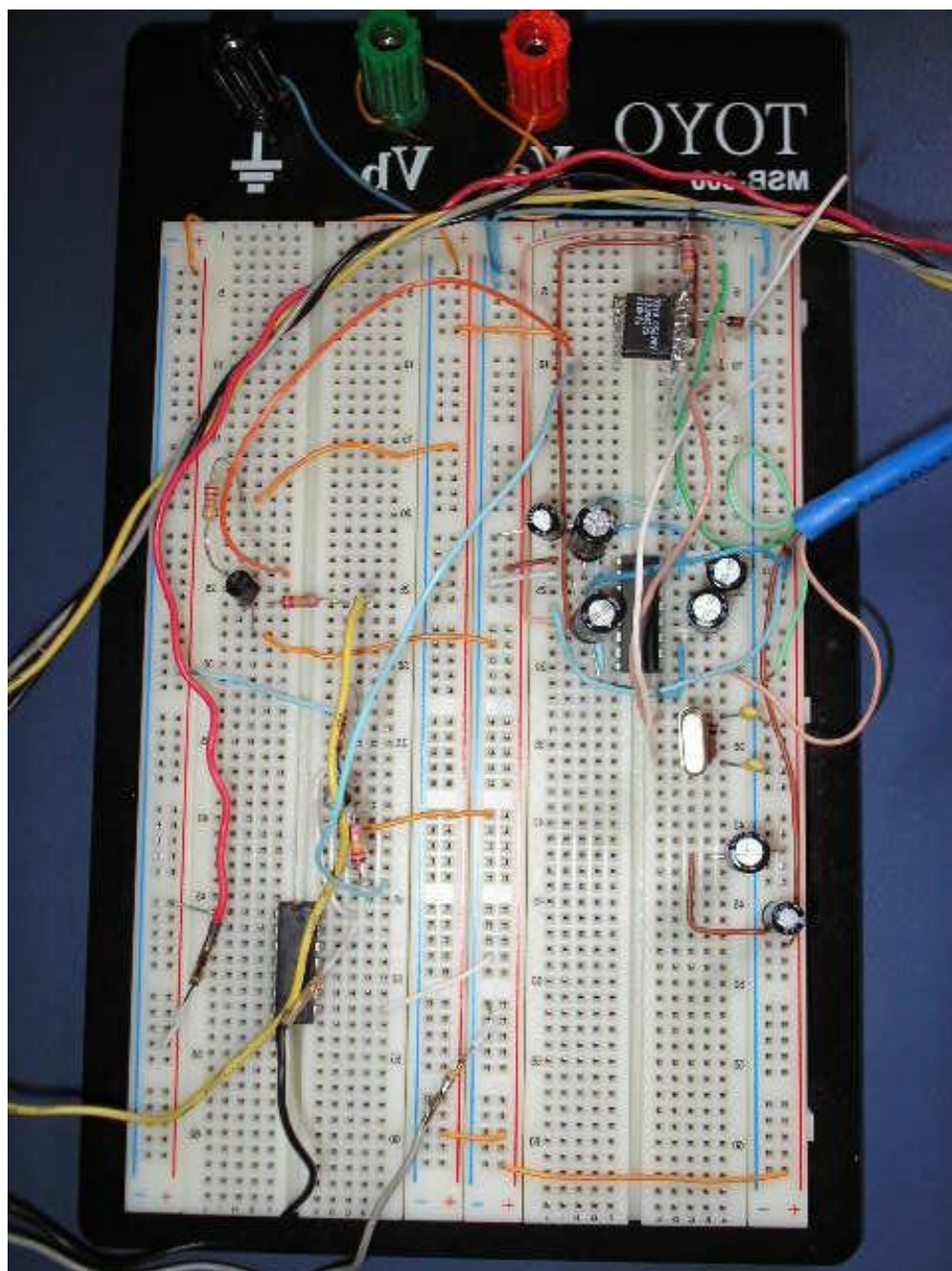


Figura 17. Placa conversora com os componentes instalados 2

### 3.1.2 Descrição TOIM4232

Este *chip* proporciona um pulso ordenado para receptores infravermelhos, sendo captados por seu *led receptor*. No modo de transmissão, o TOIM4232 diminui o

sinal de saída para o IrDA em pulsos elétricos compatível com o dispositivo de infravermelho. No modo de recepção, o TOIM4232 expande os pulsos infravermelhos recebidos, proporcionando uma maior taxa de *bits* para a operação. Sendo que a taxa de *bits* varia entre 2.4 Kbit/s e 115.2 Kbit/s.

O TOIM4232 utiliza um *clock* na faixa de 3.6864 MHz para aumentar ou diminuir o pulso por ele gerado. O *clock* pode ser gerado pelo próprio oscilador interno ou também por um *clock* externo. Este dispositivo está programado para operar entre 1200 *bit/s* até 115.200 *bit/s* na comunicação feita entre a porta RS232-C. Na Figura 18 tem-se o modelo esquemático do TOIM4232 mostrando o diagrama de bloco, onde se pode visualizar a função de cada um de seus pinos (ANEXO B).

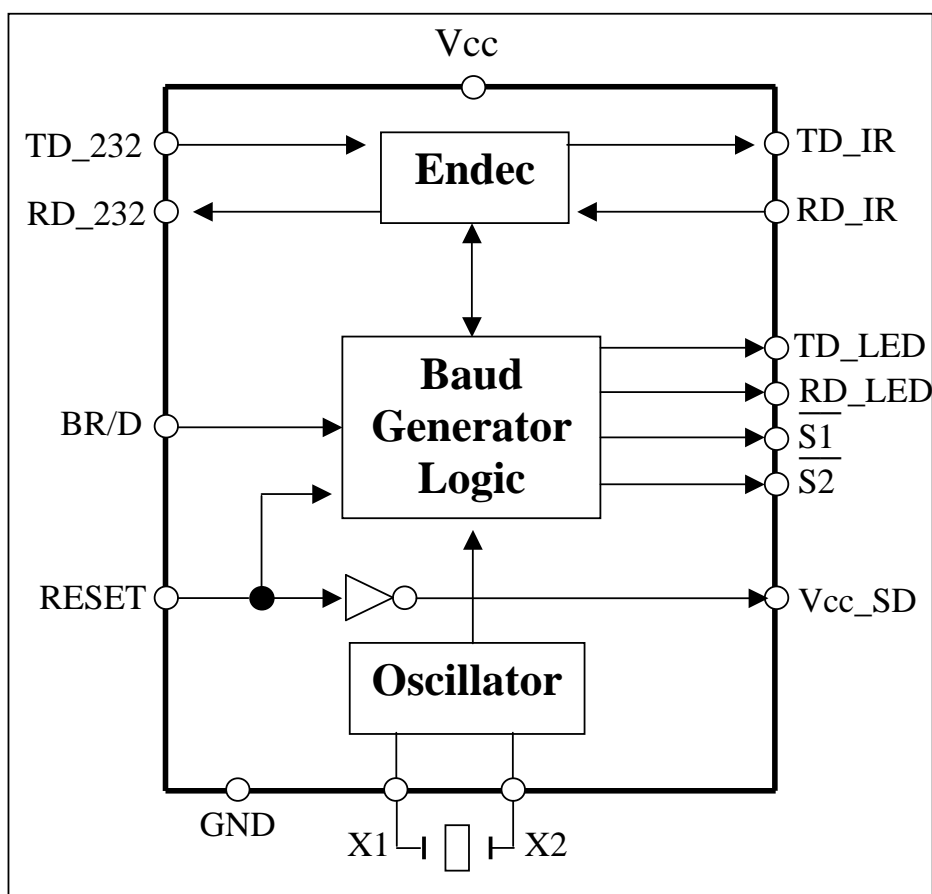


Figura 18. Diagrama de bloco TOIM4232  
 Fonte: <http://www.vishay.com>

Na Figura 19 é apresentado uma foto do circuito integrado TOIM4232.

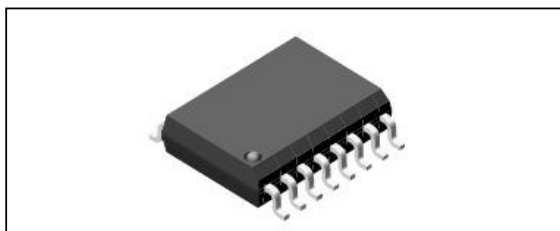


Figura 19. Foto TOIM4232  
Fonte: <http://www.vishay.com>

### 3.1.3 Descrição MAX3232E

Este componente funciona como um chaveador de tensão para interfaceamento entre os padrões RS232/IrDA, controlando e compatibilizando a saída de alimentação do dispositivo nele instalado. Consiste na regulação de energia fornecendo uma tensão de saída. A energia fornecida opera em modo descontínuo, se a voltagem de saída for menor que 5,5V a corrente será liberada, mas se a alimentação exceder a 5,5V, a alimentação é interrompida.

Na Figura 20 tem-se o MAX3232 onde pode ser visto o diagrama de bloco e visualizar a função de cada um de seus pinos.

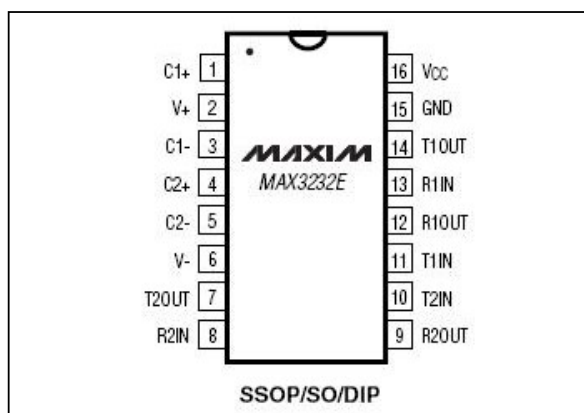


Figura 20. Diagrama de bloco MAX3232E  
Fonte: <http://www.maxim.com>

### 3.1.4 Transceiver serial infravermelho

Para o desenvolvimento do referido periférico, utilizou-se o circuito da Figura 21, onde se mostra o diagrama esquemático com seus componentes.

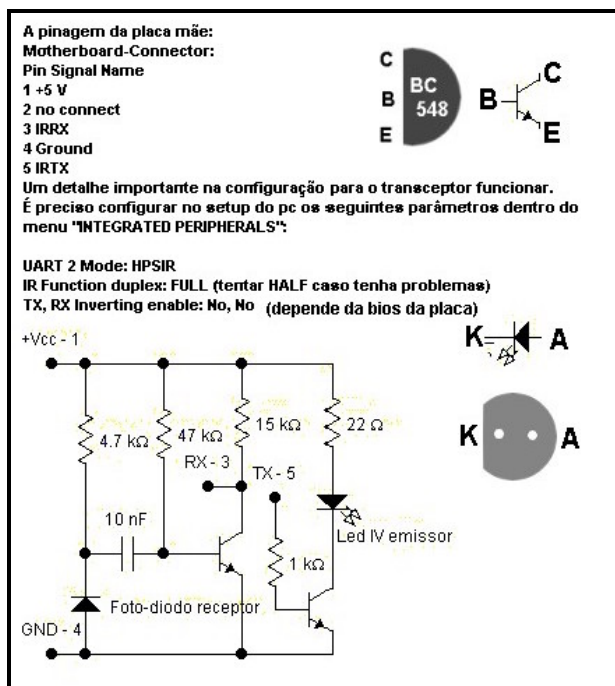


Figura 21. Modelagem do dispositivo e seus componentes  
 Fonte: <http://www.clubedohardware.com.br>

Após desenhar o *layout* de como ficariam dispostos todos os componentes na placa, foi impresso nesta mesma placa o caminho das vias condutoras para que se pudesse fazer a corrosão da mesma. Assim que a corrosão foi feita e o processo de perfuração foi finalizado, fez-se à solda de todos os componentes do circuito. A Figura 22 apresenta o *layout* da placa juntamente com as vias de cada componente utilizado e seus respectivos furos.

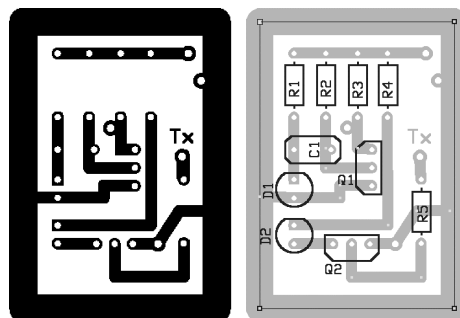


Figura 22. Desenho da placa feito no software *ExpressPCB*  
Fonte: Software *ExpressPCB*

Nela pode-se acompanhar onde cada componente será inserido. Como pode ser visto na Figura 23 os dispositivos estão soldados e alinhados conforme descrito anteriormente.

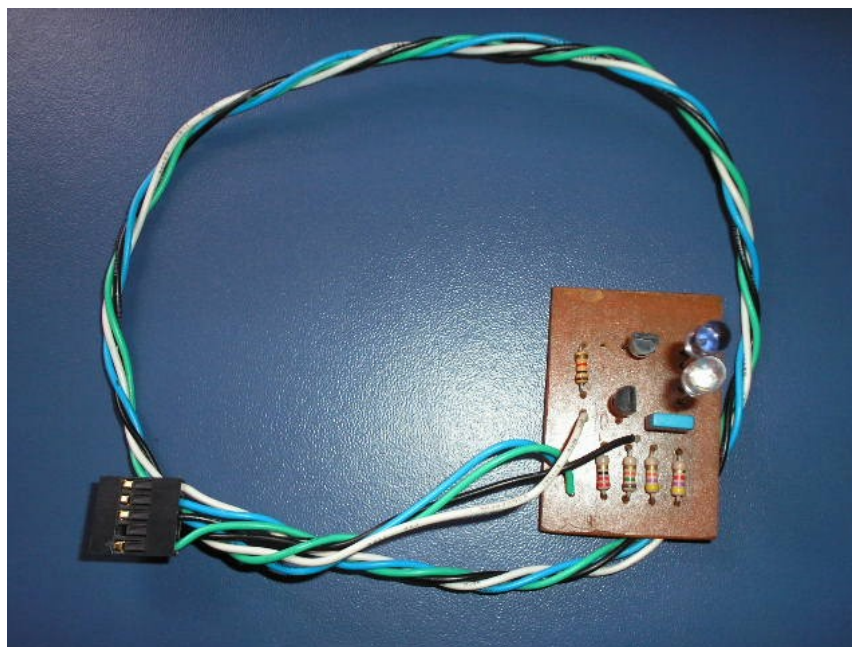


Figura 23. Placa com os componentes instalados

A seguir é descrito cada componente instalado na placa IrDA conforme mostra a Figura 24.

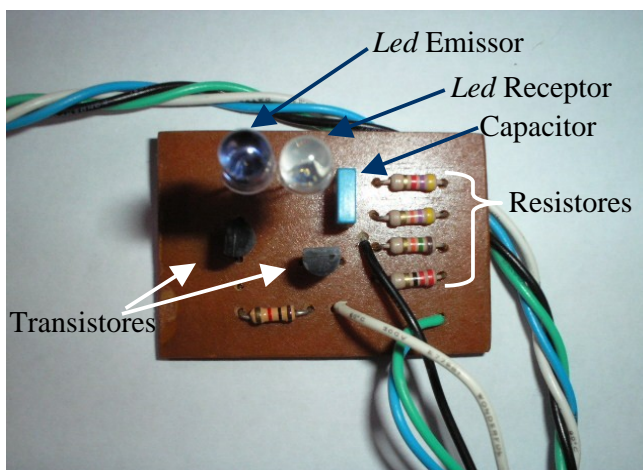


Figura 24. Placa IrDA com os componentes instalados

Na etapa seguinte, foi montado em um *protoboard* o restante do diagrama esquemático para que se faça a conversão de sinal entre o padrão IrDA e o padrão RS232-C.

### 3.1.5 Funcionamento do circuito

A partir da montagem da placa e instalação da mesma no computador, deverá ser verificado se as configurações estão de acordo com as especificações e exigências necessárias, efetuando assim a transferência dos dados. Para saber se o dispositivo está funcionando corretamente, basta aproximar os dispositivos de infravermelho a uma distância aceitável, algo em torno de 1 metro.

Quando este dispositivo se aproxima da interface a uma distância alcançável pela mesma, poder-se-á efetuar a transferência de dados entre os computadores/periféricos por meio de programa específico adequado. Nos testes realizados, foram transferidos arquivos diversos entre os computadores, diferenciando entre estes arquivos o seu tamanho para que se pudessem obter os resultados de tempo de transferência bem como estabilidade da conexão. Foram transferidas imagens, dados,

textos, programas e em todos eles a transferência atingiu níveis aceitáveis, isto é, dentro da distância limite do dispositivo.

Com relação às taxas de *bits* a que atendeu melhor a necessidade e manteve sua estabilidade foi a taxa de 9.6 *Kbps*, sendo possível manter a conexão por mais tempo sem ocorrer interferências na comunicação. É possível operar com taxas de velocidades maiores, bem como, as distâncias também. Porém, provavelmente seja necessário fazer ajustes no dispositivo para viabilizar estas especificações.

O enfoque principal foi a comunicação entre os dispositivos, podendo ser feito de forma sem fio por meio de IrDA, sendo que uma das preocupações será a segurança e a garantia desta conexão. No intuito de não utilizar nenhum meio físico para que esta comunicação ocorra, optou-se pelo infravermelho, sendo ele de grande valia para tal aplicação.

### 3.2 CASO DA COMUNICAÇÃO NAS CALCULADORAS DAS FAMÍLIAS HP 48/49/50

A comunicação de dados permite que duas calculadoras e/ou um PC transmitam um arquivo ou demais informações entre si. Porém, antes de iniciar a transferência de dados é preciso ajustar os parâmetros de comunicação da calculadora que são:

- selecionar o modo de comunicação (*Wire* ou *Infrared*);
- selecionar o protocolo de comunicação (*Kermit* ou *XModem*);
- selecionar o formato dos dados (*ASCII* ou *Binary*);
- selecionar o tipo de tradução de caracteres usado no formato *ASCII*;
- selecionar o método de verificação de erros (*checksum*) usado na transferência;

- selecionar a velocidade de transferência de dados (1200, 2400, 4800, 9600...);
- selecionar o tipo de método para gerar a paridade durante a comunicação;
- habilitar ou não a gravação (*overwrite*) de variáveis já existentes.

### 3.2.1 Protocolos de comunicação e transferência

Sabendo que os modems comunicam-se entre si para transmitir apenas *Bytes* de um lado para o outro. Eles não têm conhecimento sobre arquivos. Então para efetuar a transferência de um arquivo haverá a necessidade de se utilizar um programa/protocolo de transferência de arquivos.

Este tipo de programa/protocolo é que define a maneira que os *bytes* serão enviados e conseqüentemente interpretados do outro lado, formando assim, um arquivo exatamente de mesmo tamanho e conteúdo, igual ao que se encontra na origem. A quantidade de blocos a serem enviados será definida pelo programa/protocolo de transferência, bem como, o tratamento de erros de transmissão, sua detecção e correção, a notificação do nome do arquivo que foi enviado, informando sobre a conclusão da transferência e quantidade de arquivos que serão transmitidos.

Para isso será esclarecido mais sobre os dois protocolos que foram utilizados neste projeto são eles o programa/protocolo Kermit e o Xmodem.

#### 3.2.1.1 Protocolo de comunicação Kermit

O programa/protocolo Kermit foi desenvolvido nos anos 1990. Usado nas comunicações assíncronas foi criado pela *Columbia University*, que detém o direito

copyright, funcionando basicamente como o Ymodem (Ymodem transfere vários arquivos por conexão em fila, ou seja, ao término do primeiro arquivo continua transferindo os demais não havendo necessidade do usuário digitar o nome de cada arquivo). Sua diferença se dá pelo fato de que as partes de origem e destino definem on-line, o tamanho dos blocos a serem enviados. Sabendo da sensibilidade do Ymodem aos ruídos, isso resolveria o problema.

Havendo muito ruído em uma linha de transmissão, o Kermit reduz o tamanho dos blocos. Estabelecido este tamanho de bloco em que ocorra um valor mínimo de erros, a transmissão é iniciada. O Kermit pode ser ainda mais lento que o Xmodem quando o tamanho dos blocos é bem pequeno devido a minimizar o número de erros em uma linha ruidosa.

Uma aplicação do Kermit pode ser visto nas calculadoras gráficas, como por exemplo, HP48/49/50. Visto que as rotinas do Kermit da HP48/49/50 são lentas em função de serem escritas em *System RPL*, uma linguagem em que é escrita a maior parte da ROM da HP48/49/50. Toda vez que a HP48/49/50 recebe um pacote, interpretado como uma *string*, copia o pacote previamente recebido, definindo como *string 1* e o novo pacote definido como *string 2* para uma nova *string*, tornando-se a *string 1* quando o próximo pacote é recebido. Isto acontece em função do Kermit da HP48/49/50 ficar lento e ainda mais lento nas transferências de grandes quantidades, estabelecendo um tamanho menor para a *string 1*.

Uma curiosidade do Kermit: é o único programa/protocolo para HP48/49/50 capaz de aceitar transferência no formato ASCII, quando normalmente aceita transferência em formato binário. A universidade de Columbia detém os trustes e *copyrights* do Kermit. O nome Kermit deve-se ao Sapo Kermit (Caco, na versão brasileira) dos *Muppets Baby's*.

Kermit também é usado extensivamente na Internet, sendo um programa/protocolo antigo, possuindo várias opções que normalmente não apresentam boa performance. O tamanho dos pacotes pode variar limitando até no máximo de 1024 *bytes*.

### 3.2.1.2 Protocolo de comunicação Xmodem

O protocolo Xmodem, usado nas comunicações assíncronas, teve sua origem no ano de 1977, tendo como autor o programador chamado Ward Christensen. Ele definiu que os blocos a serem transmitidos seriam de 128 *bytes*, sendo adicionado sempre um *byte* de controle, verificando a ocorrência de erros, a esse *byte* deu-se o nome de Soma de Verificação, composto por 8 *bits* de ordem inferior a soma dos 128 *bytes*. Caso a soma dos *bytes* no destino seja diferente dos *bytes* na origem o receptor solicita uma nova transmissão dos blocos.

Seu funcionamento ocorre da seguinte forma, o Xmodem origem particiona o arquivo em blocos de 128 *bytes* cada, enviando juntamente um *byte* de *Checksum* que representa a soma dos *bytes* transmitidos. O Xmodem destino efetua a soma dos 128 *bytes* que recebeu e compara com o *Checksum*, caso não haja nenhuma diferença, envia um ACK (*acknowledgment*), que significa que o pacote foi recebido com sucesso e a origem possa enviar mais 128 *bytes*. Mas se não conferir o destino envia um NAK (*not acknowledgment*), que significa que a origem deve reenviar o bloco anterior.

Atualmente o Xmodem é considerado um protocolo pouco confiável e lento, mas foi o primeiro protocolo de larga escala a ser utilizado.

Sua forte aplicação deu-se nas calculadoras gráficas HP48/49/50 sendo aproximadamente 4 vezes mais rápido que o Kermit, apesar de ser um

programa/protocolo de transferência antigo; porém ele não transfere o nome do arquivo como acontece com o Kermit.

A pouca utilização deste protocolo deve-se ao fato de possuir 2 desvantagens:

- não permite transferências múltiplas de arquivos;
- há a necessidade de se digitar o nome do arquivo no destino.

Mas, no entanto, é muito rápido, tendo esta vantagem em transferências de arquivos que possuam tamanho maior que 5K economizando tempo e no caso das calculadoras, economia de pilha também.

### 3.3 RESULTADOS OBTIDOS

O trabalho teve como resultado, a comunicação sem fio entre dois computadores/periféricos por meio do padrão IrDA utilizando a porta serial RS232-C. Com isto, foi possível comunicar estes equipamentos e permitir a transferência de dados ou qualquer outro tipo de informação que se faça necessária.

O foco deste trabalho foi realizar uma comunicação estável que pudesse ser utilizada de modo a suprir as necessidades dos usuários. Num exemplo prático, é possível transferir qualquer tipo de arquivo que o usuário deseje sem que para isso tenha de conectar cabos entre os mesmos. Isto pode prover compatibilidade entre dispositivos com diferentes suportes de comunicação. Ou seja, dispositivos somente com interface IrDA e outros dispositivos somente com interface serial.

Foi utilizado o *Hyper Terminal* do *Windows XP* para que se pudesse realizar a comunicação proposta. Usando este recurso, foi possível verificar que em taxas muito altas de transferência de dados, como por exemplo, a 115.2 *Kbps* a conexão tornava-se

instável. Porém quando esta taxa foi definida em 9.6 *Kbps* obtinha-se estabilidade na comunicação (ANEXO C).

A distância típica para funcionamento entre os dispositivos ficou em torno de 1 metro. Porém, quanto maior à distância, menos estável a comunicação, devido aos ruídos e interferências existentes.

Testes realizados com a Calculadora Gráfica HP 48 Gii, foi possível realizar a transferência de fórmulas e sistemas do micro para a calculadora, bem como, enviar um *backup* de dados existentes na calculadora para o computador.

Uma de suas características é o baixo custo de montagem, orçado em torno de R\$ 65,00 (Sessenta e Cinco Reais) em componentes e montagem, e a não necessidade de *drivers* específicos. Basta criar uma conexão, configurada de forma igual nos dois computadores/periféricos, como por exemplo, o *Hyper Terminal* do *Windows*. Comparado com o valor de um cabo de comunicação, esse valor fica em torno da metade do preço, obtendo assim uma economia significativa.

Os resultados obtidos tanto com o sistema operacional *Windows 98* como no sistema operacional *Windows XP*, foram satisfatórios, pois atenderam as expectativas de comunicar dois periféricos utilizando o dispositivo montado.

## 4 CONCLUSÕES

Neste trabalho foi proposto a construção e a análise de um sistema de comunicação para interfaceamento do tipo IrDA-RS232-C. A taxa de transferência utilizada nos testes foi de 9.6 *Kbps*.

Observou-se que independente do sistema operacional utilizado, obteve-se bons resultados, mesmo usando um computador com o sistema operacional *Windows XP*, e outro com o sistema operacional *Windows 98*. Tornando este tipo de comunicação possível com qualquer tipo de sistema operacional, necessitando compatibilizar as configurações de protocolos entre os dois dispositivos.

O IrDA é hoje um padrão mundial em vários periféricos e equipamentos. Porém cabe ainda a compatibilidade com sistemas mais antigos. A utilização do padrão RS232-C é ainda vastamente utilizada devido à sua simplicidade .

Dentre as vantagens encontradas, pode se citar o custo total para construção do dispositivo, facilidade na aquisição dos componentes no mercado atual, fácil adaptação ao *hardware* e simplicidade na sua configuração e acesso.

Das desvantagens encontrou-se a segurança volátil, havendo a necessidade de criar algum sistema que proteja esta comunicação, não garantia da conexão estável em ambientes que haja radiação solar e a distância limitada entre os dispositivos. Deve ser observado que problemas relacionados a objetos opacos onde o infravermelho não consegue ultrapassar estabelece a interrupção da conexão.

#### 4.1 TRABALHOS FUTUROS

Como trabalhos futuros, pode-se sugerir o desenvolvimento de protocolos de segurança, protegendo desta forma, as informações a serem enviadas. Aumento da distância de alcance, entre os dispositivos, podendo assim efetuar um controle específico de algum periférico ou equipamento.

Pode-se citar também, o desenvolvimento de um *software* (programa), que utilize os protocolos de comunicação mais conhecidos, bem como, que possa permitir a troca de velocidade de comunicação. Sendo assim, seria um sistema independente de comunicação sem fio, pois estaria apto a proporcionar a comunicação entre dispositivos diferentes, por meio de um programa flexível as configurações exigidas por dispositivos externos.

## REFERÊNCIAS

BERTULANI, Carlos. **Projeto**: ensino de física a distância. Disponível em: <[www.if.ufrj.br](http://www.if.ufrj.br)>. Acesso em: 26/09/2005.

BIONDI, Sérgio. **Comunicação de dados**. Rio de Janeiro: Brasport, 1996.

HELD, Gilbert. **Comunicação de dados**. 6.ed. Rio de Janeiro: Campus, 1999.

KNUTSON, Charles D.; BROWN, Jeffrey M. IrDA principles and protocols: the IrDA library. v.1. Salen (USA): MCL Press, 2004.

MARTINS, Rosilda Baron. **Metodologia científica**: como tornar mais agradável a elaboração de trabalhos acadêmicos. Curitiba: Juruá, 2004.

SILVEIRA, Jorge Luiz da. **Comunicação de dados e sistemas de teleprocessamento**. São Paulo: Mc Graw Hill, 1991.

STALLINGS, William. **Data & computer communications**. 6.ed. New Jersey: Prentice Hall, 1997.

ZELENOVSKY, Ricardo; MENDONÇA, Alexandre. **PC um guia prático de hardware e interfaceamento**. 2.ed. Rio de Janeiro: MZ, 1999.

**BIBLIOGRAFIA RECOMENDADA**

ALENCAR, Marcelo Sampaio de. **Sistemas de comunicações**. São Paulo: Érica, 2001.

BEZERRA, Eduardo Augusto. **Conectando sistemas embarcados**. Faculdade de Informática, PUC-RS. Porto Alegre, mar. 2004. Disponível em: <[http://toledo.inf.pucrs.br/~eduardob/disciplinas/topicos\\_sdacii\\_pos/NotasAula8/com\\_p\\_ara\\_alela.html](http://toledo.inf.pucrs.br/~eduardob/disciplinas/topicos_sdacii_pos/NotasAula8/com_p_ara_alela.html)>. Acesso em: 12/08/2005.

DORNAN, Andy. **Wireless communication**: o guia essencial de comunicação sem fio. Rio de Janeiro: Campus, 2001.

HAMACHER, V. Carl; VRANESIC, Zvonko G.; ZAKY, Safwat G. **The computer organization**. Estados Unidos da America: The MCGraw-Hill, 1996.

HENNESSY, John L.; PATTERSON, David A.; MACHADO FILHO, Nery. **Organização e projeto de computadores**: a interface hardware/software. 2.ed. Rio de Janeiro: LTC, 2000.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Fundamentos de metodologia científica**. 6.ed. São Paulo: Atlas, 2005.

LIMA JÚNIOR, Almir Wirth. **Tecnologias de redes & comunicação de dados**. Rio de Janeiro: Alta Books, 2002.

MÁTTAR NETO, João Augusto. **Metodologia científica na era da informática**. 2.ed. São Paulo: Saraiva, 2005.

SETUBAL, Rogério. **WAP wireless application protocol do servidor ao site**. Rio de Janeiro: Book Express, 2000.

STUHLMULLER, P. Mini-Web Server: Fundamentos da comunicação pela internet. **Revista Eletrônica & Microinformática**. Ano 1, n.3, jun. 2002.

TANENBAUM, Andrew S. **Organização estruturada de computadores**. 3.ed. Rio de Janeiro: Prentice-Hall do Brasil Ltda, 2000.

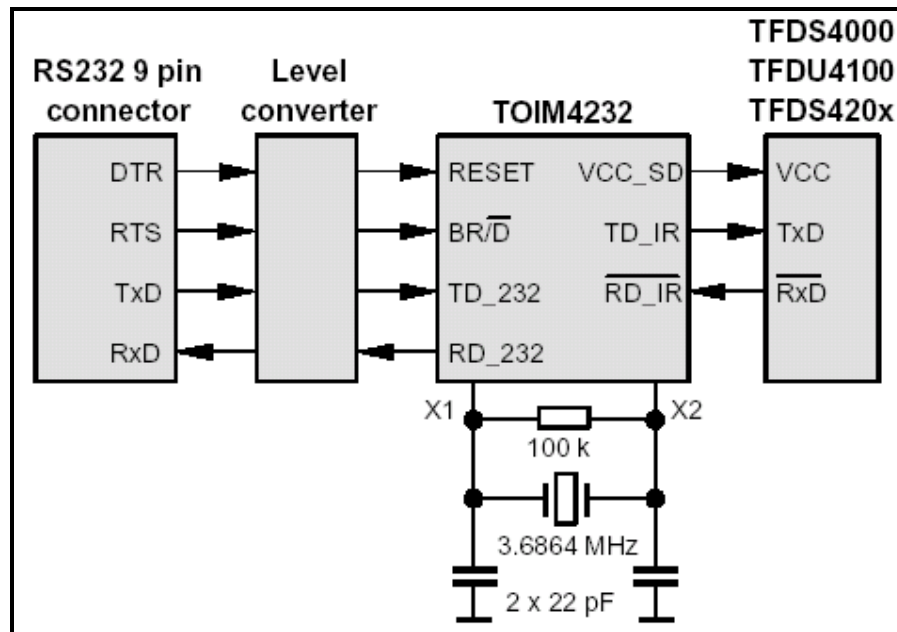
TORRES, Gabriel. **Hardware curso completo**. 3.ed. Rio de Janeiro: Axcel Books, 2000.

VASCONCELOS, Laércio; ASSUMPÇÃO FILHO, Milton Mira de. **Hardware total**. São Paulo: Makron Books, 2002.

ZUFFO, João Antonio. **Microprocessadores**: dutos do sistema, técnicas de interface e sistemas de comunicação de dados. São Paulo: Edgard Blücher, 1981.

**ANEXOS**

**ANEXO A – MODELO DE APLICAÇÃO DO CIRCUITO UTILIZANDO  
PORTA RS232-C (ADAPTADOR EXTERNO INFRAVERMELHO)**



Fonte: <http://www.vishay.com>.

## ANEXO B – INDICAÇÃO DOS PINOS E DESCRIÇÃO TOIM4232

### Pin Assignment and Description

Pin Number	Symbol	Description	I/O	Active
1	RESET	Resets all internal registers. Initially must be HIGH ("1") to reset internal registers. When HIGH, the TOIM4232 sets the IrDA default bit rate of 9600 bit/s, sets pulse width to 1.627 $\mu$ s. The V <sub>CC_SD</sub> output is simply an inverted reset signal which allows to shut down of a TFDx400 transceiver when applying the reset signal to the TOIM4232. When using devices with external SD like TFD54203, the reset line can be used directly as shut down signal. RESET pin can be controlled by either the RTS or DTR line through RS232 level converter. Minimum hold time for resetting is 1 $\mu$ s. Disables the oscillator when active.		HIGH
2	BR/ D	<b>Baud Rate control/ Data.</b> <b>BR/ D = 0, data communication mode:</b> RS232 TXD data line is connected (via a level shifter) to TD_232 input pin. The TXD - signal is appropriately shortened and applied to the output TD_IR, driving the TXD input of the IR transceiver. The RXD line of the transceiver is connected to the RD_IR input. This signal is stretched to the correct bit length according to the programmed bit rate and is routed to the RS232 RXD line at the RD_232 pin. <b>BR/ D = 1, Programming mode:</b> Data received from the RS232 port is interpreted as Control Word. The Control Word programs the baud rate which will be effective as soon as BR/ D return to LOW.		
3	RD_232	Received signal data output of stretched signal to the RS232 RXD line (using level converter).	O	HIGH
4	TD_232	Input of the signal to be transmitted from the RS232 port TXD line (passing the level converter).	I	HIGH
5	V <sub>CC_SD</sub>	Outputs an inverted RESET signal. Can be used to shut down the power supply of a 4000 series transceiver (e.g., TFDU4100). V <sub>CC</sub> shutdown output function. This pin can be used to shut down a transceiver (e.g., TFDx4xxx). Output polarity: inverted RESET input.	O	LOW
6	X1	Crystal input clock. 3.6864 MHz nominal. Input for external clock *)	I	
7	X2	Crystal *)	I	
8	GND	Ground in common with the RS232 port and IrDA transceiver ground		
9	TD_LED	Transmit LED indicator driver. Use 180 $\Omega$ current limiting resistor in series to LED to connect to V <sub>CC</sub> (V <sub>CC</sub> = 3.3 V)	O	LOW
10	RD_LED	Receive LED indicator driver. Use 180 $\Omega$ current limiting resistor in series to LED to connect to V <sub>CC</sub> (V <sub>CC</sub> = 3.3 V)	O	LOW
11	NC	No connection		
12	S1	User Programmable Bit. Can be used to turn ON/ OFF a front-end infrared transceiver (e.g., an infrared module at the adapter front)	O	LOW
13	S2	User Programmable Bit. Can be used to turn ON/ OFF a front-end infrared transceiver (e.g., an infrared module at the adapter back)	O	LOW
14	TD_IR	Data output of shortened signal to the infrared transceiver	O	HIGH
15	RD_IR	Data input from the infrared transceiver, min. pulse duration 1.63 $\mu$ s **)	I	LOW
16	VCC	Supply voltage	I	

\*) Crystal should be connected as shown in figure 2. In addition connect a 100 k $\Omega$  resistor from Pin 6 to Pin 7 and from Pin 6 and Pin 7 a 22 pF capacitor to ground, respectively. When an external clock is available connect it to Pin 6 leaving Pin 7 open. The external resistor of 100 k $\Omega$  is used to accelerate the start of oscillation after reset or power - on. The value depends on the "Q" of the resonator. With low Q resonators it is not necessary. The start - up time of the oscillator is between 30  $\mu$ s (with piezo resonators) and above 2 ms with high Q quartzes.

\*\*) All Vishay Semiconductor SiR transceivers fulfill this condition

Fonte: <http://www.vishay.com>.

### ANEXO C – TAXA DE TRANSFERÊNCIA(\*)

B3	B2	B1	B0	2nd Char	Baud Rate
0	0	0	0	0	115.2 k
0	0	0	1	1	57.6 k
0	0	1	0	2	38.4 k
0	0	1	1	3	19.2 k
0	1	0	0	4	14.4 k
0	1	0	1	5	12.8 k
0	1	1	0	6	9.6 k
0	1	1	1	7	7.2 k
1	0	0	0	8	4.8 k
1	0	0	1	9	3.6 k
1	0	1	0	A	2.4 k
1	0	1	1	B	1.8 k
1	1	0	0	C	1.2 k

\* O padrão IrDA suporta somente os seguintes valores: 2.4, 9.6, 19.2, 57.6 e 115.2 kbit/s.  
 Fonte: <http://www.vishay.com>.