



Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Curso de Graduação em Engenharia Elétrica

RAMSÉS ARAUJO GONÇALVES

**SISTEMA DE AQUISIÇÃO DE DADOS ATRAVÉS DE
INTERFACE DE COMUNICAÇÃO IRDA**

Campina Grande, Paraíba
Outubro de 2013

RAMSÉS ARAUJO GONÇALVES

SISTEMA DE AQUISIÇÃO DE DADOS ATRAVÉS DE INTERFACE DE COMUNICAÇÃO IRDA

*Trabalho de Conclusão de Curso submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Controle e Automação/Transmissão de dados

Orientador:

Professor Alexandre Cunha Oliveira, Dr.

Campina Grande, Paraíba
Outubro de 2013

RAMSÉS ARAUJO GONÇALVES

SISTEMA DE AQUISIÇÃO DE DADOS ATRAVÉS DE INTERFACE DE COMUNICAÇÃO IRDA

*Trabalho de Conclusão de Curso submetido à Unidade
Acadêmica de Engenharia Elétrica da Universidade
Federal de Campina Grande como parte dos requisitos
necessários para a obtenção do grau de Bacharel em
Ciências no Domínio da Engenharia Elétrica.*

Área de Concentração: Controle e Automação/Transmissão de dados

Aprovado em / /

Professor Marcos Ricardo Alcântara Morais, Dr.
Universidade Federal de Campina Grande
Avaliador

Professor Alexandre Cunha Oliveira, Dr.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho aos meus pais, João e Verônica, exemplos de dignidade, amor e caráter, que com tanto esforço conseguiram realizar o sonho de ver todos os seus três filhos formados.

AGRADECIMENTOS

Agradeço primeiramente a Deus por todas as oportunidades que me proporcionou, por todas as dificuldades, obstáculos e experiências vividas, servindo tudo isto como aprendizagem e crescimento.

Agradeço também a toda a minha família, nas figuras de meus pais João e Verônica, além dos meus irmãos, Radamés e Rayanne que a todo tempo se fizeram presentes, e puderam me auxiliar em muitos momentos, sendo com palavras de apoio ou mesmo com auxílio na minha mobilidade, sendo todos eles peça chave para mais essa conquista.

Agradeço também a pessoa da minha namorada Monailza, por toda força e compreensão, estando sempre ao meu lado, me apoiando, me erguendo muitas vezes, acompanhando toda a minha trajetória na graduação e que com as bênçãos de Deus me acompanhará para o resto da vida.

Agradeço também aos meus irmãos de curso Kennedy, André (Alemão), Dayvson, Wesley, Diogo, Pablo e Leiva além de muitos outros que não foram citados devido ao tamanho elevado que iria ficar essa relação. Com eles passei muitas noites em claro, compartilhando muitos conhecimentos, desejos, perspectivas, sonhos, decepções, conquistas, refeições insalubres, e é por causa do nosso esforço coletivo que hoje estamos conseguindo terminar este curso que mais parece uma prova de resistência.

Agradeço por fim aos meus muitos professores por terem dedicado muito do seu tempo em prol de que possamos cada vez mais consolidar saberes e com isso fortalecer progressivamente nosso corpo discente e, por conseguinte o nome da nossa instituição. Cito em especial as figuras dos professores Antônio Marcus, Marcos Morais, Alexandre Cunha, Waslon Terllizzie, Damásio Fernandes, George Lira entre outros, como exemplos do potencial do nosso departamento. Cada um destes marcou minha vida na graduação servindo como referencial de conhecimento, junção de habilidades prática e intelectual, multidisciplinaridade, convivência, metodologia e didática além de inovação e o empenho de fazer sempre o melhor.

Enfim, agradeço a todos que de alguma forma, passaram pela minha vida e contribuíram para a construção de quem sou hoje.

*“A alegria está na luta,
na tentativa, no
sofrimento envolvido e
não na vitória
propriamente dita.”*

Mahatma Gandhi

RESUMO

Ao longo dos anos o homem vem procurando realizar melhorias no seu padrão de vida seja com relação a sua comodidade, a sua saúde, segurança, em fim. O que se percebe é a constante evolução tecnológica transformando o meio. Computadores, impressoras, televisões, vídeo games, aparelhos de som, centrais de ar, todos fazem parte da rotina e estes na grande maioria possuem controle remoto que é o principal representante do processo de comunicação por infravermelho, sendo este meio de comunicação a fonte de análise desse projeto. [3]

Na atualidade, é comum a procura por televisões e dispositivos de alta definição, os quais possuem um tráfego considerável quando se observa a transferência de informações, devido ao volume de dados gerado, o que pode se tornar um processo demorado. Para este fim um grupo de cientistas conseguiu desenvolver um padrão de comunicação por infravermelho de alta velocidade alcançando atualmente a taxa de 1Gbps. [11]

Este projeto visa analisar, especificar e detalhar como funciona o padrão de comunicação por infravermelho IrDA. Além disso, será discutido a figura do microcontrolador e por fim realizar-se-á parte prática com a leitura e verificação de erros de comunicação de um dispositivo IrDA.

Palavras-chave: Comunicação wireless, IrDA, Microcontroladores, PIC.

ABSTRACT

Over the years man has been trying to realize improvements in their standard of living is relative to your comfort, your health, safety, in order. What is noticeable is the constant technological changes transforming the medium. Computers, printers, TVs, video games, stereos, central air, all part of the routine and these in most feature remote control that is the main representative of the communication process by infrared, and this means of communication the power analysis of this project.

Nowadays, it is common to search for high-definition TVs and devices, which have a considerable traffic when observing the transfer of information, due to the volume of data generated, which can become a lengthy process. To this end a group of scientists has succeeded in developing a communication standard for high speed infrared currently increasing the rate of 1Gbps.

The purpose of the project present here is to analyze, specify and detail how the IrDA infrared standard communication protocol works. In addition, aspects about microcontroller technology are also discussed. Finally, as proof of concept, the experimental set-up is built for reading and error checking in IrDA communication.

Keywords: Wireless Communication, IrDA, Microcontrolers, PIC.

LISTA DE ILUSTRAÇÕES

Figura 1 – Espectro de Luz.....	3
Figura 2 – Representação da Pilha de protocolos IrDA.....	9
Figura 3 – Representação do Sinal IrDA Serial.....	12
Figura 4 – Pacote de dados da comunicação serial síncrona.....	12
Figura 5 – Diagrama representativo do microcontrolador.....	14
Figura 6 – Famílias PIC e o tamanho dos barramentos (dados e instruções).....	15
Figura 7 – Representação da Arquitetura Harvard do PIC 16F877A.....	16
Figura 8 – Diagrama em Blocos do PIC16F877A.....	17
Figura 9 – Pinagem e Invólucro DIP para o PIC16F877A.....	18
Figura 10 – Pinagem e Invólucro SMD (TQFP) para o PIC16F877A.....	18
Figura 11 – Circuito Básico para PIC16F877A.....	19
Figura 12 – Aplicação de interface IrDA com microcontrolador.....	20
Figura 13 – Esquemático de adaptador IrDA recomendado pela Maxim.....	21
Figura 14 – Conexão de módulo transceptor IrDA com o computador.....	21
Figura 15 – Diagrama de aplicação IrDA sugerido pela mikroe.....	22
Figura 16 – TFDU 4300 (diagrama sugerido na documentação IrDA).....	22
Figura 17 – Comparação no formato da palavra Serial e da IrDA.....	23
Figura 18 – Processo de conversão IrDA/RS232 (compensação da largura do pulso com temporizador).....	23
Figura 19 – Esquema do Adaptador do Sinal IrDA para RS232.....	24
Figura 20 – Análise da resposta do oscilador monoestável 555 ao disparo.....	24
Figura 21 – Montagem do Adptador IrDA/RS232.....	25
Figura 22 – Simulação Conversor IrDA/Serial.....	26
Figura 23 – Fluxograma do programa do Conversor IrDA/Serial.....	27
Figura 24 – Parte do código do conversor Serial/IrDA para a detecção do bit de start	28
Figura 25 – Linhas de código referente a implementação do sinal IrDA.....	28
Figura 26 – Linhas de código relacionadas a configuração do Timer 2.....	29
Figura 27 – Análise comparativa entre o byte escrito e o byte lido após processo de conversão.....	30
Figura 28 – Linhas de Código referente ao tratamento da interrupção do Timer 2.....	30
Figura 29 – Pinos da porta IrDA na placa mãe.....	32

LISTA DE ABREVIATURAS E SIGLAS

A/D – Analógico/Digital

ALU – Arithmetic Logic Unit

Checksum – Soma de verificação

EEPROM – Electrically Erasable Programmable Read-Only Memory
(Memória Programável eletricamente apagável)

FIR – Fast Infrared (Infravermelho Rápido)

I2C – Inter-Integrated Circuit

IAS – Information Access Service (Serviço de Acesso a Informação)

IrDA – Infrared Data Association (Associação de Dados por Infravermelho)

IrLAP – Link Access Protocol (Protocolo de Acesso de Ligação)

IrLMP – Link Management Protocol (Protocolo de Gerenciamento de Ligação)

MIR – Medium Infrared (Médio Infravermelho)

PWM – Pulse Width Modulation (Modulação por Largura de Pulso)

RAM – Random Access Memory (Memória de Acesso Aleatório)

RISC – Reduced Instruction Set Computer (Computador com Set de Instruções Reduzidos)

SIR – Serial Infrared (Infravermelho Serial)

Smart Meters – Medidores Inteligentes

SPI – Serial Peripheral Interface (Periférico de Interface Serial)

ULA – Unidade Lógica Aritmética

USART – Universal Synchronous Asynchronous Receiver Transmitter
(Transmissor-Receptor Universal Síncrono-Assíncrono)

VFIR – Very Fast Infrared (Infravermelho Muito Rápido)

SUMÁRIO

1	Introdução	1
1.1	Objetivos	1
1.2	Motivação	2
1.3	Estrutura do Trabalho	2
2	Comunicação por Infravermelho - IrDA	3
2.1	Visão Geral	3
2.2	A Organização	4
2.3	Vantagens e Desvantagens.....	6
2.3.1	Vantagens ao se utilizar a IrDA	6
2.3.2	Desvantagens no padrão IrDA	7
2.4	Características	7
2.5	Detalhamento da Plataforma.....	8
2.5.1	Protocolos Requeridos.....	9
2.5.2	Protocolos Opcionais.....	10
2.5.3	A Camada Física	11
2.6	Aplicações e Desenvolvimentos	13
3	Microcontrolador PIC	14
3.1	Características do PIC 16F877A.....	15
3.2	Estrutura Interna do PIC 16F877A	16
3.3	Requisitos de Hardware	18
4	Aplicação Prática	20
4.1	Dispositivos e circuitos IrDA recomendados	20
4.2	Simplex Conversor IrDA/RS232	22
4.3	Leitor IrDA unidirecional com PIC	26
5	Conclusão.....	31
6	Sugestões para Trabalhos Futuros	32
	Bibliografia.....	33
	APÊNDICE A – Programa do Conversor IrDA/Serial (Simulação no Proteus).....	36
	APÊNDICE B – Programa do Conversor Serial/IrDA (Simulação no Proteus).....	39

1 INTRODUÇÃO

Já ocorreram situações onde ao se esquecer de um pequeno cabo conector tem-se a impossibilidade de utilização de determinado dispositivo o que causa frustração e faz pensar como seria se não necessitasse daquele cabo para efetuar a comunicação. Pensando nesse propósito e na integração dos dispositivos por IR surgiu o padrão IrDA.

Fundada em 1993 por um grupo de empresas, a IrDA, abreviação de Infrared Data Association, é uma interface de comunicação sem fio que se utiliza da luz infravermelha como meio de propagação dos dados, podendo na sua camada física ser classificada como uma comunicação serial similar ao padrão RS232. [11]

Semelhante a comunicação serial padrão, a IrDA apresenta taxas de transmissão padronizadas, além de bits de início e de fim para cada byte transmitido, se diferenciando na característica do nível do sinal (forma que se é representado o bit) e apresentando alta imunidade a ruído, sendo assim utilizada em computadores, celulares, impressoras, calculadoras e em diversos outros equipamentos.

Para realizar a leitura dos dados provenientes de uma porta IrDA fazendo verificação de erros por meio de checksum, será utilizado o microcontrolador.

Microcontroladores são dispositivos programáveis compostos de uma CPU e diversos periféricos como memórias, conversores analógico/digitais, timers, entre outros, acoplados em um único encapsulamento. Por meio de linhas de código pode-se controlar suas entradas e saídas e gerenciar diversos processamentos, além de realizar inúmeras tarefas necessitando apenas da criatividade do usuário em utilizar estes dispositivos e aliá-los a outras tecnologias. [6], [7]

1.1 OBJETIVOS

Além de abordar os avanços na área da comunicação sem fio utilizando infravermelho, este projeto visa também o estudo análise e implementação de um sistema microcontrolado para leitura, registro e transferência de informações obtidos por meio de equipamentos que possuam uma porta de comunicação ótica IrDA (Infrared Data Association).

1.2 MOTIVAÇÃO

Em meio ao avanço das imagens de alta resolução e do crescente volume de dados a serem transferidos entre diferentes dispositivos tem-se buscado tecnologias que possam suprir necessidades de altas taxas de transferências sem que fosse necessária a utilização de cabos ou conectores. Nesse contexto o padrão IrDA vem se destacando. As tecnologias de transmissão por infravermelho vêm ganhando mercado como resultado de um intenso trabalho e cooperação entre companhias industriais de renome. [11]

Este trabalho vem pra desmistificar a ideia passada por muitos autores, os quais veem a IrDA como uma tecnologia ultrapassada, considerando o Bluetooth como seu sucessor.[3],[17]

1.3 ESTRUTURA DO TRABALHO

Este trabalho pode ser dividido em três partes, sendo as duas primeiras mais teóricas enquanto que a última apresenta um carácter mais prático.

Inicialmente faz-se neste trabalho uma abordagem com o foco no padrão IrDA, explicitando suas características, detalhando suas vantagens e desvantagens, retratando onde se faz a utilização desta tecnologia partindo para uma análise geral dos protocolos adotados dando um destaque maior a camada física.

Em seguida tem-se o alvo da pesquisa voltado para os microcontroladores, em especial para o modelo do PIC16F877A, podendo-se observar detalhes como arquitetura interna, softwares utilizados além de requisitos de hardware necessários para a execução de atividades com estes componentes.

Posteriormente faz-se uma abordagem mais prática por meio de um apanhado de circuitos e diagramas sobre a utilização da IrDA em computadores e em microcontroladores seguidos de uma descrição de dois pequenos projetos com o enfoque nesse tema, sendo o primeiro dotado de características mais analógicas enquanto que o último utiliza as funcionalidades da programação de microcontroladores.

O trabalho é finalizado com o encaminhamento das conclusões e sugestões para trabalhos futuros.

2 COMUNICAÇÃO POR INFRAVERMELHO - IRDA

No cotidiano, é notória a presença cada vez maior de produtos tecnológicos como computadores, celulares, aparelhos de som, impressoras e até mesmos em equipamentos mais comuns como geladeiras, purificadores de água, cafeteiras e televisões, é crescente a inserção de tecnologias para dar maior flexibilidade e opções aos seus usuários. Muitos desses dispositivos aliam a possibilidade de acionamentos e customizações das suas funções por meio de comandos enviados por uma central ou por qualquer outro dispositivo que possua alguma conexão com o aparelho. Neste cenário, por permitir uma melhor estética pela redução e muitas vezes até desaparecimento de fios e condutores, além de reduzir problemas de conexão devido a diferenças de plugues e conectores, as tecnologias sem fio se destacaram e se mostram cada vez mais fortes e atrativas. [11]

Dentre as tecnologias sem fio existentes, tem-se a comunicação por infravermelho, sendo o controle remoto o dispositivo mais popular ao utilizar predominantemente este meio de transmissão, tendo sua utilização em televisores, dvd players, aparelhos de som, sistemas de iluminação, apresentadores de slides, entre outros. [3]

2.1 VISÃO GERAL

Para compreender o funcionamento da tecnologia de comunicação por infravermelho, analisemos primeiro a característica do espectro de luz para este fenômeno.

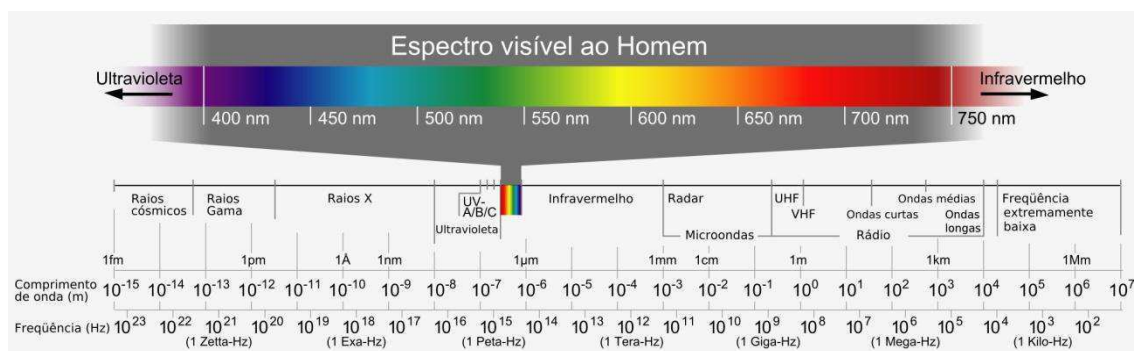


Figura 1 – Espectro de Luz

A radiação por infravermelho, cujo significado do termo é “abaixo do vermelho”, foi descoberta em 1800, pelo astrônomo inglês de origem alemã William Herschel, compreendendo a faixa entre 700 nm e 50 000 nm. É classificada como uma radiação não ionizante, já que esse tipo de radiação é de baixa energia de forma que os átomos de uma determinada substância submetidos a esta faixa de frequência podem vibrar sem provocar reação. [16]

O controle remoto é o maior representante da transmissão de dados por infravermelho, sendo este responsável pelo envio de dados, muitas vezes palavras binárias, para os dispositivos a serem controlados, caracterizando na grande maioria uma comunicação unidirecional. Para que fosse possível a bidirecionalidade do canal de comunicação por infravermelho, se fez necessário o uso de sistemas ou medidas complexas, o que fez com que cada fabricante montasse uma tecnologia que melhor atendesse as suas especificidades acarretando em incompatibilidade entre dispositivos de diferentes marcas. Partindo desta premissa, após a reunião de um grupo de empresas, em 1993 fundou-se a Infrared Data Association (IrDA), cuja tarefa era a de formular, promover e desenvolver padrões a serem seguidos para dispositivos de comunicação por infravermelho. [5], [11]

2.2 A ORGANIZAÇÃO

Sendo uma organização de âmbito internacional, a IrDA, sediada em Walnut Creek, Califórnia, possui como gestores um conselho de administração que representa os interesses de mais de 160 membros corporativos em todo o mundo. [9], [20]

Como diretoria executiva da IrDA tem-se uma equipe encabeçada atualmente pela pessoa de Gontaro Kitazumi da E-Globaledge, sendo o trabalho da organização dividido em três frentes. Têm-se os comitês: Técnico, de Marketing e o de Testes e Operações, sendo eles responsáveis pelo refinamento, identificação, inovação e ampliação dos padrões IrDA, pelo desenvolvimento de medidas promocionais estratégicas e táticas de mercado, além da criação de documentos, ensaios e procedimentos que garantam a conformidade de equipamentos e dispositivos protegendo e assegurando o fortalecimento da marca respectivamente. [21]

Dentre as empresas que participam da elaboração dos padrões e do desenvolvimento de novas tecnologias voltadas a este seguimento, tem-se:

- Access
- ACTiSYS
- Assn for Retail Technology Standards(ARTS)
- Assn Interactive Media
- Casio Computer
- Compaq
- Computer Engineering and Consulting
- Control A/D
- E-Globaledge
- Fraunhofer IPMS
- Freescale
- FSTC
- Fujifilm
- Fujitsu
- iAnywhere Solutions
- KDDI CORPORATION
- Lattice Semiconductor
- Liteon
- Motorola
- Nokia
- Panasonic
- Paycircle
- PDAia
- Rohm
- Sharp Electronics
- Sony Ericsson Mobile Communications AB
- Systeklabs
- Toshiba
- Vishay
- Waseda University Test Lab. [12]

2.3 VANTAGENS E DESVANTAGENS

Assim como em toda tecnologia, a IrDA apresenta inúmeras vantagens e também suas limitações e é no pesar dessa relação que se verifica o potencial desse meio de comunicação.

2.3.1 VANTAGENS AO SE UTILIZAR A IRDA

Segundo a página oficial da IrDA, pode-se pensar nessa tecnologia como sendo, de baixo custo em relação as demais formas de comunicação wireless, uma forma segura de substituição de cabos de forma conveniente e excelente para muitas aplicações e ambientes específicos, além de ser desenvolvida por muitos indivíduos e qualificada por centenas de empresas que, ao longo dos anos, compunham a Infrared Data Association. [11]

Dentre esses aspectos, pode-se citar como vantagens na utilização da IrDA:

- Menor custo em relação a outras tecnologias. [9]
- Possui atualmente tecnologias de rápida comunicação (desenvolvimento de padrões que obtém marcas de transmissão de 1, 5 e 10Gbps (128MBps a 1.25GBps)). [11]
- Baixo consumo de energia (em muitos dispositivos pode-se optar pela transmissão com baixa potência garantindo uma redução de 10 vezes no consumo padrão, porém não se podem exceder distâncias maiores que 20 centímetros entre os dispositivos). [9], [15]
- Segurança dos dados e informações devido à direcionalidade da transmissão, sendo útil em transferência de informações em operações financeiras e outros tipos de informações sigilosas, impossibilitando a interceptação de terceiros.
- Permite maior controle e consciência das conexões estabelecidas.
- Trata-se de um padrão mundialmente adotado e implementado em mais de 1 bilhão de produtos em todo o mundo.

- Padrões IrDA são não-proprietários. Por serem abertos, as implementações IrDA promovem cada vez mais o padrão e torna possível a interoperabilidade entre plataformas, marcas e tipo de produto, se devidamente implementada.
- IrDA é relativamente livre de regulamentação. Frequências infravermelhas são aquelas encontradas um pouco abaixo da luz visível na escala eletromagnética não havendo restrição na sua utilização já que esta não produz interferências em frequências de rádio ou conflitos de sinal com outras aplicações. Por não promover problemas este tipo de tecnologia é especialmente importante em ambientes críticos como hospitais e aeroportos. [11]

2.3.2 DESVANTAGENS NO PADRÃO IRDA

- Limitação na distância entre dispositivos, ou seja, um curto alcance, não ultrapassando um ou dois metros para transmissão de dados, e sete metros para transmissão de comandos ou informações de controle.
- Dificuldade em manter a comunicação sem que haja uma espécie de visada direta entre os dispositivos não ultrapassando o limite de 15° de inclinação.
- Dificuldade para a comunicação simultânea para mais de um dispositivo. [2]
- Muitos computadores não vem com a porta IrDA conectada ou instalada, possuindo na placa mãe apenas um dos possíveis conectores definidos pelo padrão. [4]

2.4 CARACTERÍSTICAS

Além de detalhamento de hardware, as especificações dos padrões IrDA possuem uma série de documentos relacionados aos protocolos a serem utilizados pelos dispositivos implementados, podendo-se dividir em duas famílias de aplicações: a de dados e a de controle.

A IrDA de Dados, como a própria denominação sugere, segue padrões para a conexão entre dispositivos para transferência de arquivos ou dados. Cria-se uma rede ponto-a-ponto com distâncias entre dispositivos que não ultrapassem 1 metro de comprimento e velocidades que vão desde 2400bps a 16Mbps para aplicações mais comuns.

A família de especificações para a IrDA de Controle consiste de orientações voltadas à comunicação (envio de pequenos pacotes) entre periféricos sem fio tais como mouse, teclado, controle remoto, apresentadores de slides, e um Host remoto como um PC, ou outro dispositivo de consumo, promovendo a interação entre estes dispositivos. Este padrão trata com faixas de valores de até 7 metros de distância entre os dispositivos e taxas de transmissão que podem atingir 75kbps. [9], [18]

Têm-se como características atuais do padrão IrDA as seguintes generalidades:

- Conexão sem fio universal.
- Ampla gama de hardware suportada em diversas plataformas.
- Substituição dos cabos por uma rede ponto-a-ponto.
- Compatibilidade com versões anteriores do padrão (legado).
- Pequena angulação (30 graus) em forma de cone no raio de ação entre dispositivos.
- Aplicativos estilo point-and-shoot, não interferindo com outros equipamentos eletrônicos.

2.5 DETALHAMENTO DA PLATAFORMA

Utilizando as vantagens de uma estruturação consolidada, a IrDA possui inúmeras especificações contemplando todas as camadas de abstração do Modelo OSI, podendo ainda dividir os protocolos do padrão IrDA em dois tipos, os padrões requeridos e opcionais.

IrDA Stack

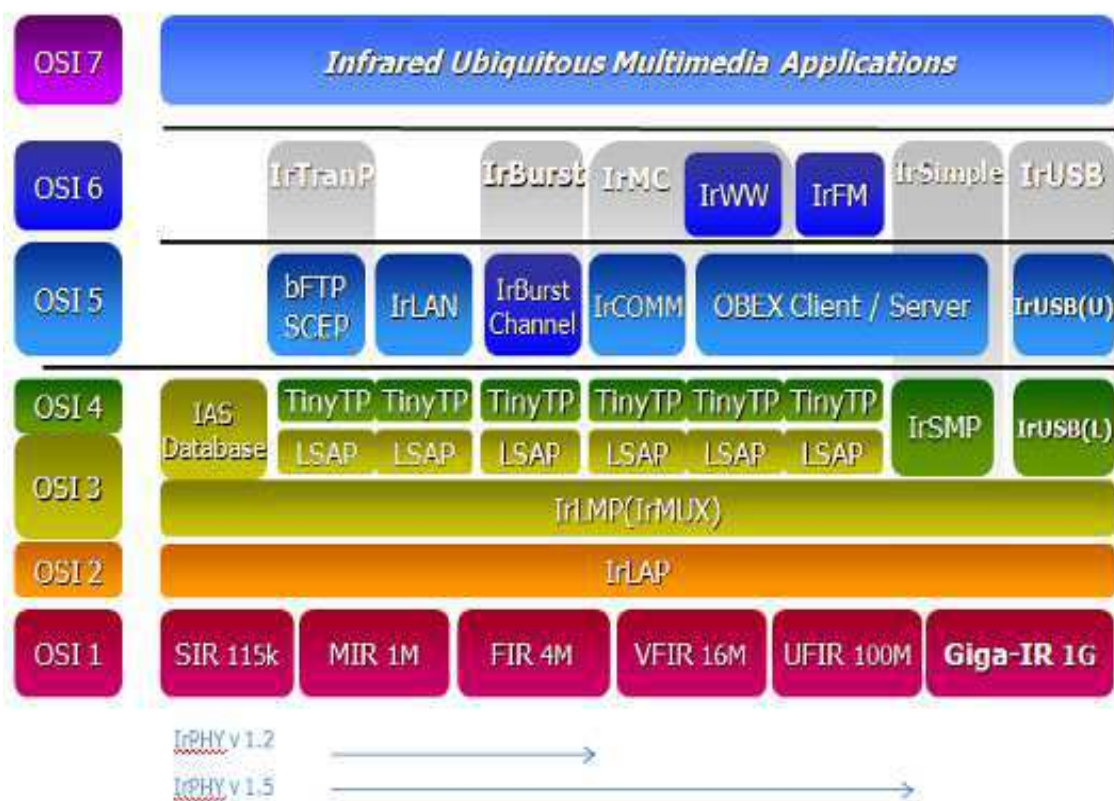


Figura 2 – Representação da Pilha de protocolos IrDA

2.5.1 PROTOCOLOS REQUERIDOS

Têm-se os seguintes protocolos requeridos:

Physical Layer: Especifica características óticas, codificação de dados e o suporte a várias velocidades e/ou taxas, além de regras para CRC, codificação de bits de dados, entre outras.

IrLAP (Link Access Protocol): Estabelece uma conexão básica confiável, coletando informações sobre outros dispositivos, possibilitando a escolha de um dentre outros acessíveis. Por meio de uma sondagem, parametriza-se e cria o link, transferindo os dados quando possível e quando não, alertam-se as camadas superiores.

IrLMP (Link Management Protocol): Podendo ser dividido em duas partes, realiza a multiplexação de serviços e aplicações na conexão fornecida pelo LAP, permitindo que vários cliente se comuniquem.

IAS (Information Access Service): Provém informações sobre protocolos e serviços, consistindo sua implementação de clientes e servidores onde o cliente busca pela informação nos servidores.

2.5.2 PROTOCOLOS OPCIONAIS

Têm-se os seguintes protocolos opcionais:

Tiny TP: Oferece controle de fluxo nas conexões IrLMP com uma segmentação opcional e serviço de remontagem, dividindo grandes pacotes durante o envio e remontando-os no recebimento. Sendo implementada na terceira camada do modelo OSI, adiciona um byte de informações em cada pacote IrLMP para realizar suas tarefas.

IrCOMM: Fornece uma comunicação virtual seja ela serial ou paralela para aplicações anteriores que utilizavam esses meios de comunicação. Trata-se da emulação de portas seriais e paralelas sobre a pilha de protocolos IrLMP / IrLAP. A motivação para IrCOMM vem da variedade de aplicações de impressão e de comunicação que usam APIs de comunicação padrão para se comunicar com outros dispositivos através de portas seriais e paralelas. IrCOMM emula portas seriais RS-232 (EIA/TIA-232-E), e portas paralelas Centronics, como as encontrados na maioria dos computadores pessoais.

IRFM: Protocolo desenvolvido para permitir um sistema de pagamento digital que irá reduzir os custos de transação para o comerciante e as instituições financeiras além de fornecer um método mais simples de rastreamento de transações financeiras entre indivíduos e empresas. Os custos totais vão cair com o uso de sistemas de pagamento sem fio, o que irá beneficiar comerciantes e consumidores.

OBEX™: Trata-se de um protocolo eficaz compacto, binário que permite uma troca de dados em uma grande variedade de dispositivos de uma maneira simples e espontânea. No entanto, OBEX não se limita a transferência rápida, ele também permite que sessões nas quais transferências ocorram ao longo de um período de tempo, mantenham-se conectadas, mesmo quando estejam inativas. PCs, pagers, PDAs, telefones, impressoras, câmeras, auto-caixas, quiosques de informação, calculadoras, dispositivos de coleta de dados, relógios, eletroeletrônicos, máquinas industriais, instrumentos médicos, automóveis e equipamentos de escritório são todos candidatos a usarem OBEX.

IrDA Lite: Fornece métodos para reduzir o tamanho do código IrDA, mantendo a compatibilidade com implementações completas.

IrSimple: Alcança maiores velocidades de transmissão de dados (pelo menos de 4 a 10 vezes mais rápido do que as atuais, melhorando a eficiência do atual protocolo IrDA infravermelho embutido em muitos dispositivos móveis, como telefones celulares. Além disso, o protocolo IrSimple também mantém a compatibilidade com os protocolos IrDA existentes. Possui como características principais desse padrão a velocidade de transmissão de dados mais rápida (pelo menos de 4 a 10 vezes mais rápido que os protocolos existentes).

IrTran-P: Fornece protocolo de troca de imagem utilizadas em dispositivos digitais como câmeras entre outras.

IrMC: Especificações sobre como dispositivos de telefonia e comunicações móveis possam trocar informações. Isso inclui a agenda, calendário e dados de mensagens. A norma define medidas para uma fácil troca de cartões de visita, notas, itens de calendário e listas de tarefas entre os dispositivos móveis, definindo também o controle de chamadas e transferência de áudio entre um telefone celular e um computador ou notebook.

IrLAN: Descreve um protocolo usado para apoiar o acesso sem fio IR para redes locais.

IrWW: Especificação para relógios de pulso. [13]

2.5.3 A CAMADA FÍSICA

Existem basicamente três especificações no padrão para métodos de transmissão a nível físico de acordo com o modo de transmissão e com a velocidade requerida: [21]

SIR (Serial Infrared) Assíncrono: Para velocidades de 2400bps até 115200bps. Utiliza o mesmo formato de dados da porta serial - transmissão assíncrona de dados com um start bit, um stop e 8 bits de dados. Utiliza o chip padrão de comunicação serial UART para transmitir os dados. A figura 3 ilustra o formato dos dados;

2.6 APLICAÇÕES E DESENVOLVIMENTOS

A interface de comunicação por infravermelho pode ser encontrada em: PDAs, notebooks, desktops, pagers, telefones celulares, impressoras, relógios, dispositivos médicos, eletrodomésticos, smart meters, doc stations, entre outros. [5]

Além destas aplicações mais populares outros segmentos de pesquisa e desenvolvimento do padrão estão em constante evolução.

Relógios com dispositivos IrDA para automação residencial estão sendo utilizados. Com eles é possível ligar e desligar luzes, sistemas de alarme, controlar a central de ar condicionado, manipular com eletrodomésticos como uma espécie de controle universal, observar o batimento cardíaco para idosos e hipertensos entre outras aplicações.

Dispositivos de controle de acesso também estão sendo aplicados em Parques como no Bungalow Park, onde o visitante recebe uma espécie de relógio que permite a ele fazer reservas, observar detalhes de algo específico como programações e eventos, além de funcionar como identificador. [22]

Para museus podem ser utilizados os IRádios, uma espécie de rádio que permite ao visitante conhecer detalhes de obras e esculturas por meio do direcionamento do dispositivo ao feixe emitido por circuitos nas obras de arte. Esta ideia seria interessante de ser utilizada em cafés e lanchonetes nos estádios, durante a copa do mundo no Brasil.

Com o avanço das tecnologias de alta definição, os usuários estão cada vez mais exigentes e com isso a capacidade de armazenamento e o volume dos dados transferidos estão cada vez maiores o que demanda melhorias nos meios de comunicação adotados. Seguindo esta linha, grupos de pesquisadores estão desenvolvendo padrões e circuitos de alta velocidade para a transmissão de dados por infravermelho, tendo o projeto a denominação de Giga-IR. Atualmente já se obteve taxas de 1Gbps utilizados por circuitos próprios conectados a Smartphones e dispositivos de áudio utilizados tanto para sincronização de informações, quanto para transferência de arquivos maiores como vídeos entre outros. [11]

A ideia é de num futuro próximo não serem mais necessários mídias digitais para compra de filmes, shows e jogos, bastando que o usuário posicione seu dispositivo pessoal e em poucos segundos ele terá todo o conteúdo que ele desejar.

3 MICROCONTROLADOR PIC

Microcontroladores são dispositivos programáveis compostos de uma pequena CPU e diversos periféricos como memórias, conversores analógicos/digitais, temporizadores, PWMs, entre outros, acoplados em um único encapsulamento. [10]

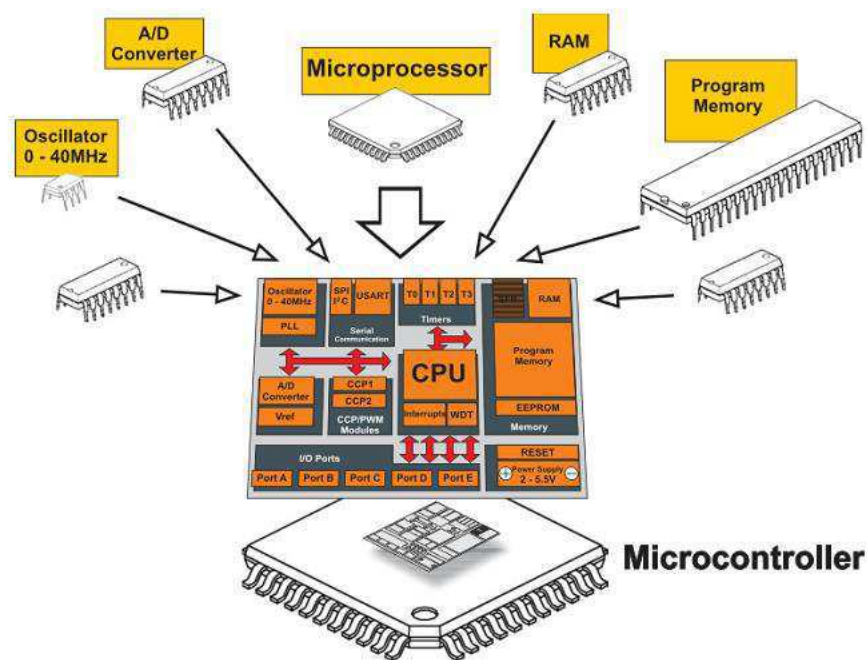


Figura 5 – Diagrama representativo do microcontrolador.

Por meio de linhas de código podem-se controlar suas entradas e saídas e gerenciar diversos processamentos e realizar inúmeras tarefas necessitando apenas da criatividade do usuário em utilizar estes dispositivos e aliá-los a outras tecnologias.

São encontrados microcontroladores em aparelhos de DVD, televisões, celulares, calculadoras, vídeo-games, sistemas de alarmes, circuitos de automação residencial, portões de garagem, automóveis, PDAs, eletrodomésticos, equipamentos médicos, entre outros. [10]

Empresas como Microchip, Atmel, Philips, Motorola, Psoc, Zilog, Holtek, entre outras, investem no campo dos microcontroladores, dotando seus dispositivos de periféricos e circuitos voltados a diferentes necessidades surgindo assim às famílias geralmente distinguidas pelas capacidades de processamento e armazenamento.

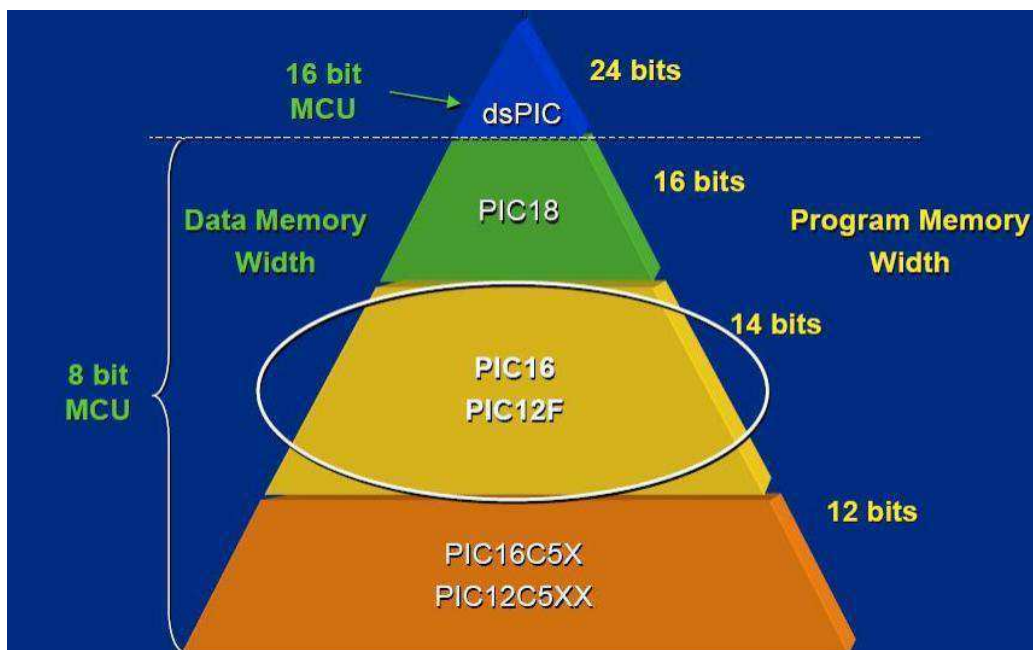


Figura 6 – Famílias PIC e o tamanho dos barramentos (dados e instruções).

As Famílias 16 e 18 de microcontroladores PIC se mostram bastante versáteis e flexíveis a aplicações de baixa complexidade além de fáceis de serem obtidas no mercado nacional possuindo assim muito material informativo devido a sua popularidade. A programação destes dispositivos pode ser realizada em Assembly ou em C, e esta tarefa pode ser realizada de forma menos onerosa por meio da utilização de bibliotecas de componentes. [6]

3.1 CARACTERÍSTICAS DO PIC 16F877A

Como características do PIC 16F877A tem-se uma composição de 40 pinos do tipo DIP ou 44 pinos na versão SMD, conjunto de instruções reduzido (RISC) com 35 instruções, arquitetura Harvard com uma memória de programa e outra de dados, pilha implementada por hardware com 8 níveis, suporte a cristal oscilador de até 20 MHz, alimentação de 2 a 5,5V, memória de programa do tipo Flash de 8192 palavras de 14 bits, 368 bytes de memória RAM para dados, 256 bytes de memória EEPROM para dados, 33 terminais distribuídos em 5 portas que podem ser configurados como entrada e/ou saída, conversor analógico-digital de 10 bits de resolução e 8 canais de entrada multiplexados, 2 Módulos CCP (Comparação, Captura e PWM), porta paralela com 8 bits de dados e sinais de controle externos (RD, WR e CS), 14 fontes de interrupção entre externas e internas, programação in-circuit serial, proteção por código, modo

sleep para operação com baixo consumo de energia, 3 timers (2 de 8 bits e 1 de 16 bits), Watchdog timer (temporizador de segurança), power-on-reset, power-up timer, portas seriais síncronas com SPI e I2C, além de USART e MSSP;

3.2 ESTRUTURA INTERNA DO PIC 16F877A

Por apresentarem uma arquitetura Harvard, tendo a separação entre a memória de dados e a memória de programa, as instruções podem ser representadas por palavras maiores que 8 bits, permitindo assim que as instruções ocupem uma só palavra, como é o caso do PIC16F877A cujo barramento de instruções utiliza um total de 14 bits como representado na figura abaixo. [6], [10]

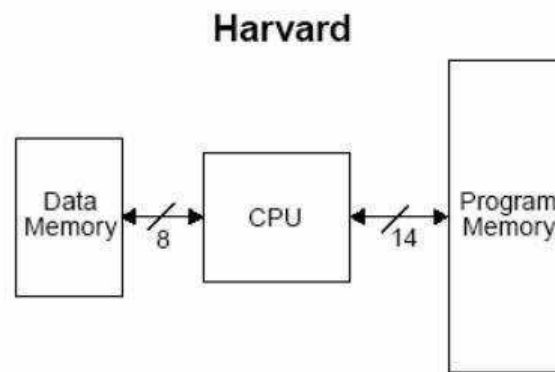


Figura 7 – Representação da Arquitetura Harvard do PIC 16F877A.

Este tipo de arquitetura permite um processamento mais rápido já que as memórias de dados e de programa por meio dos barramentos de dados e de instruções podem ser acessadas de forma concorrente. [10]

Outra característica primordial para as famílias PIC a ser destacada é o fato da utilização da filosofia RISC contendo apenas 35 instruções para o PIC16F877A, o que pode deixar algumas tarefas maiores, porém o diferencial se apresenta na rapidez da execução, de forma que pode-se obter uma velocidade de 5 MIPS (milhões de instruções por segundo) com um cristal de 20MHz para este modelo. Desta forma um ciclo de máquina equivale a quatro ciclos do cristal oscilador obtendo assim uma relação direta entre o valor do cristal oscilador e o tempo decorrido para cada instrução.

Devido à arquitetura do PIC, tem-se também a capacidade de pipeline, fazendo com que enquanto se executa uma instrução se faz o carregamento da instrução seguinte, além da execução de uma instrução por ciclo de máquina para instruções

simples e de dois ciclos para instruções de desvio, tendo ainda que cada instrução ocupa apenas uma posição na memória de programa e com isso obtém-se também um tempo fixo para a execução da instrução.

Pela representação da figura abaixo se pode ter uma visão macro da estrutura do microcontrolador PIC 16F877A.

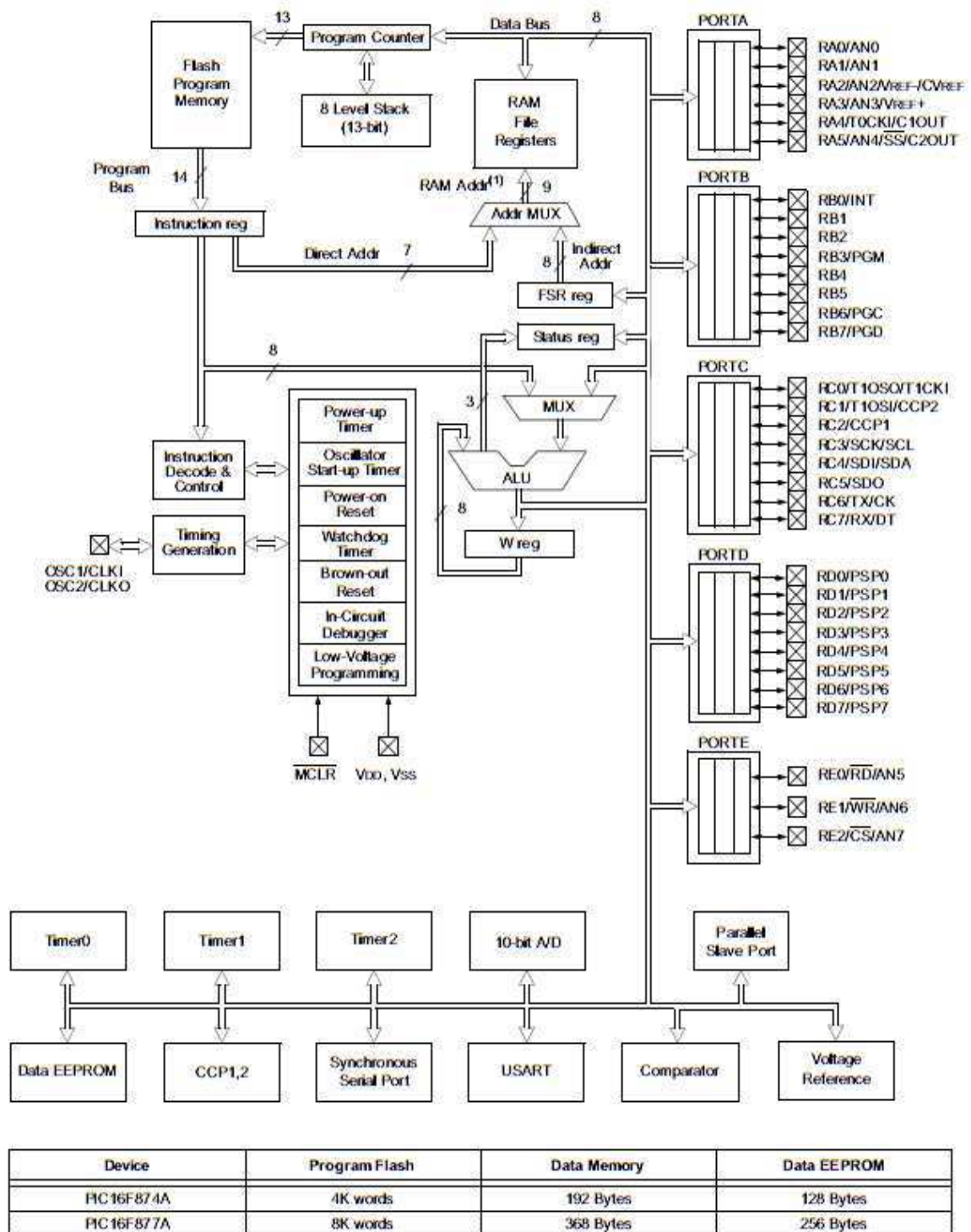


Figura 8 – Diagrama em Blocos do PIC16F877A.

3.3 REQUISITOS DE HARDWARE

O PIC 16F877A pode ser encontrado com 3 tipos de invólucros diferentes, sendo os dois modelos representados nas figuras abaixo como os mais comuns seja por facilidade na utilização em protoboards para testes e experimentos, ou mesmo por simplificar o processo de soldagem manual.

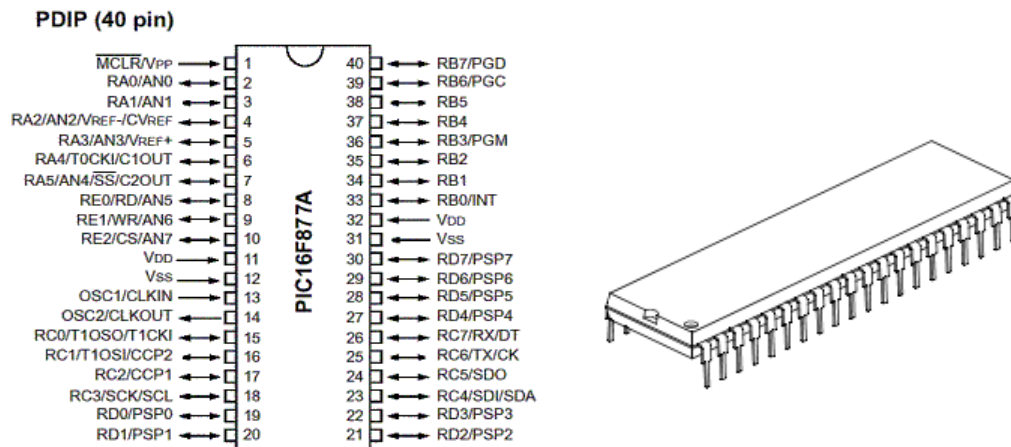


Figura 9 – Pinagem e Invólucro DIP para o PIC16F877A.

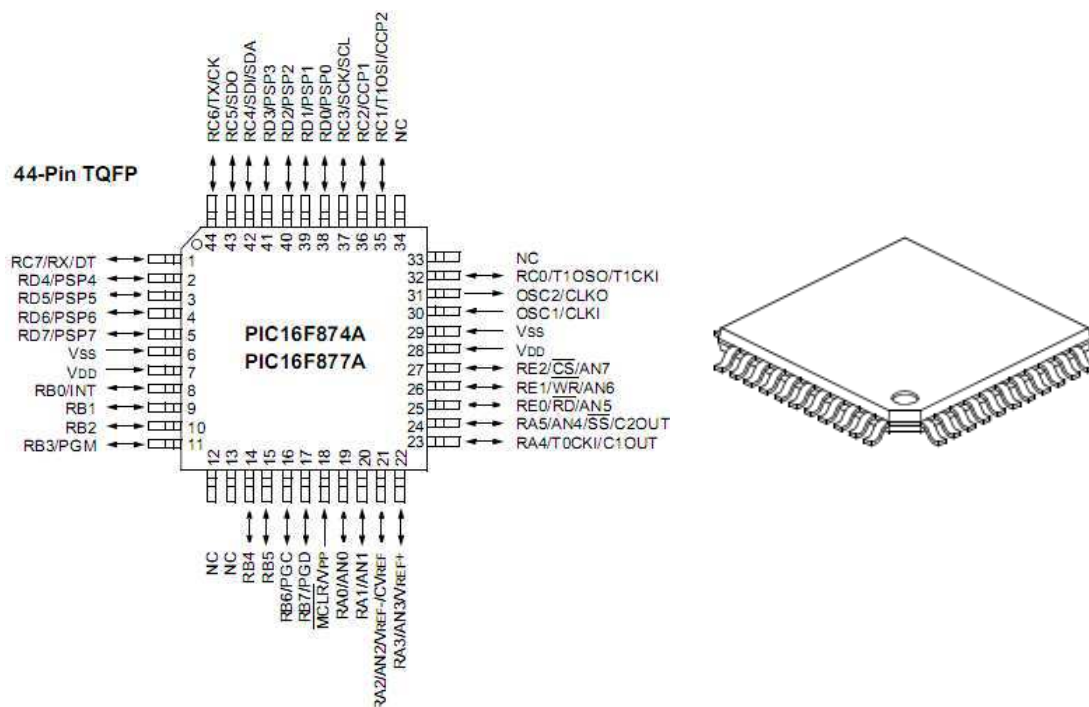


Figura 10 – Pinagem e Invólucro SMD (TQFP) para o PIC16F877A.

Mesmo possuindo um oscilador interno, recomenda-se a utilização de um cristal externo como gerador de clock para o microcontrolador garantindo assim uma maior precisão no funcionamento do componente, principalmente quando a aplicação envolve a utilização de temporizadores e comunicação serial cujas atividades estão diretamente associadas a este parâmetro.

Assim como para todo circuito é interessante à utilização de uma boa fonte de alimentação cujo regulador de tensão tem a finalidade de dar maior estabilidade e evitar problemas inesperados como reinicializações indevidas ou mal funcionamento de periféricos.

Na figura abaixo segue o esquema básico de montagem para a utilização do PIC16F877A.

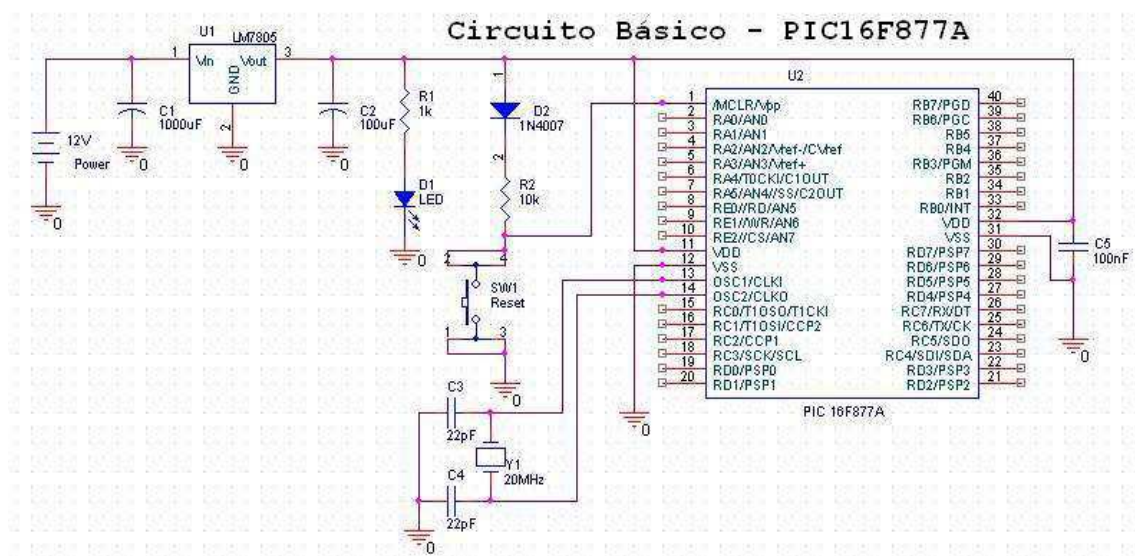


Figura 11 – Circuito Básico para PIC16F877A.

Para escrever os códigos-fonte para o PIC e obter os arquivos hexadecimais necessários para a gravação no dispositivo, o programador pode optar por softwares como o MPLab da Microchip e o PICC da CCS (PCWH) para plataforma windows, ferramentas essas capazes de gerenciar projetos, emular, simular e compilar os programas. [6], [10]

Para gravar os programas compilados no microcontrolador se faz necessário a utilização de um circuito gravador e de um software de gravação. A própria Microchip disponibiliza esquemáticos de gravadores pela denominação de ProgPic e é possível obter gravadores conectados ao computador pelas portas usb, serial e paralela. Para a gravação pode-se utilizar os softwares WinPIC800, PicKit, Picstart Plus, McFlash entre outros disponibilizados para esta finalidade. [10]

4 APLICAÇÃO PRÁTICA

4.1 DISPOSITIVOS E CIRCUITOS IRDA RECOMENDADOS

Por apresentar um padrão singular, sendo caracterizado como um caso particular da RS232, existem alguns circuitos integrados que realizam a tarefa de conversão dos sinais IrDA, adequando-os a diferentes formatos de comunicação a serem adotados. Entre eles tem-se o MCP2120 que é um codificador/decodificador de comunicação por infravermelho, possuindo pinos de seleção de baud rate e de modo de operação podendo-se optar entre modo de dados e de comandos. Para a transmissão e recepção do sinal por infravermelho tem-se o HSDL1100, podendo funcionar nas faixas de 2,4 a 115,2Kbit/s pelo canal de saída primário e dos 576Kbit/s a 4,0Mbit/s utilizando o canal secundário.[23]

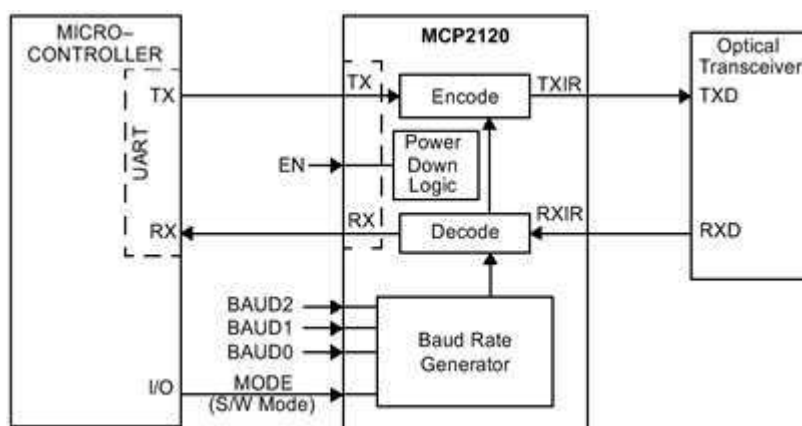


Figura 12 – Aplicação de interface IrDA com microcontrolador.

A operação conjunta entre os circuitos integrados MAX3100 e MAX3131 também pode ser utilizada como solução para a utilização do padrão IrDA. O primeiro circuito integrado é responsável por gerar uma oscilação de referência a ser utilizada pelo MAX3131, que nada mais é que um transceptor IrDA/RS232, podendo esta oscilação ser ajustada por meio de comandos enviados pela porta de comunicação SPI do dispositivo gestor do processo. [24]

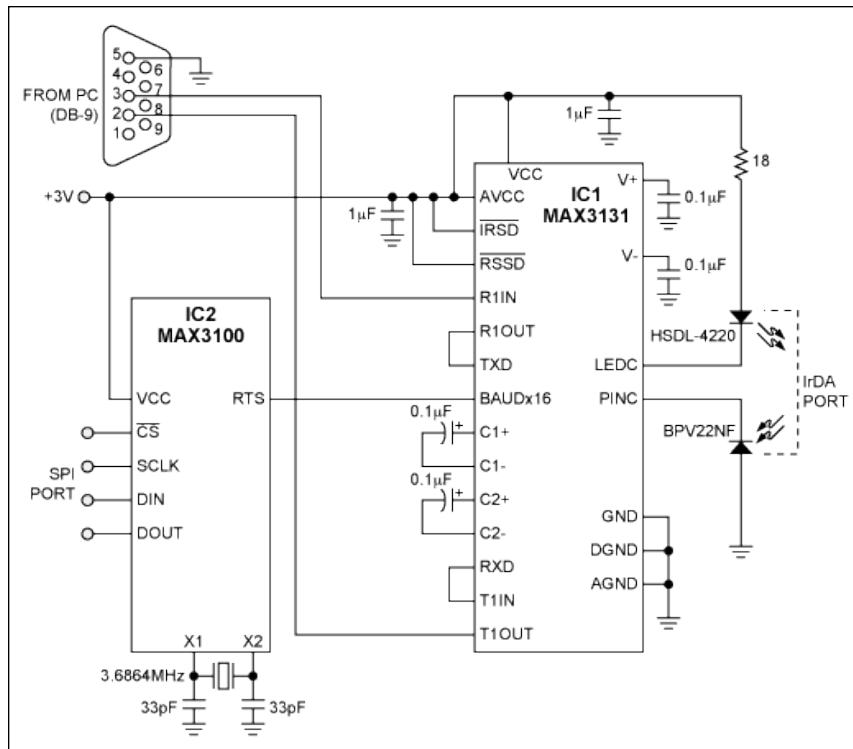


Figura 13 – Esquemático de adaptador IrDA recomendado pela Maxim.

Caso o dispositivo a ser utilizado na comunicação infravermelha com o padrão IrDA seja um computador, é provável que este possua circuitos para esta aplicação na própria placa mãe, necessitando apenas que o usuário realize as conexões corretas do circuito transceptor como pode ser observado na figura 14.

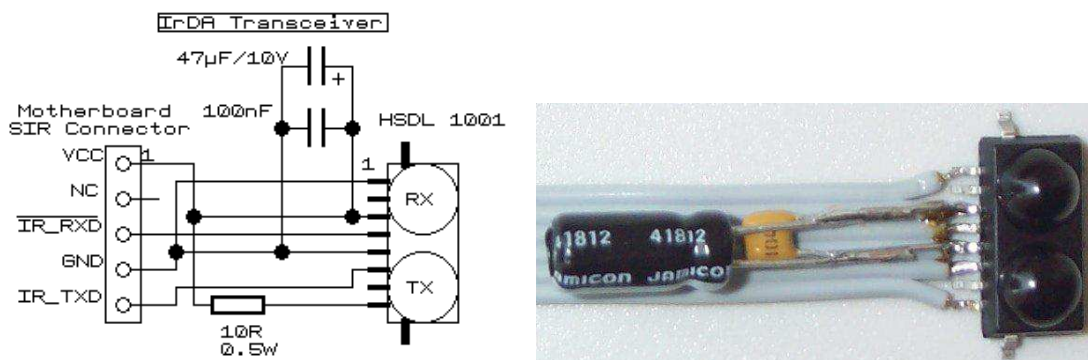


Figura 14 – Conexão de módulo transceptor IrDA com o computador.

Para esta montagem descrita, o circuito integrado para o transceptor de infravermelho foi o HSDL1001 da Agilent Technologies. [25]

Também podem ser utilizados os circuitos integrados MCP2155 como codificador/decodificador IrDA. [19]

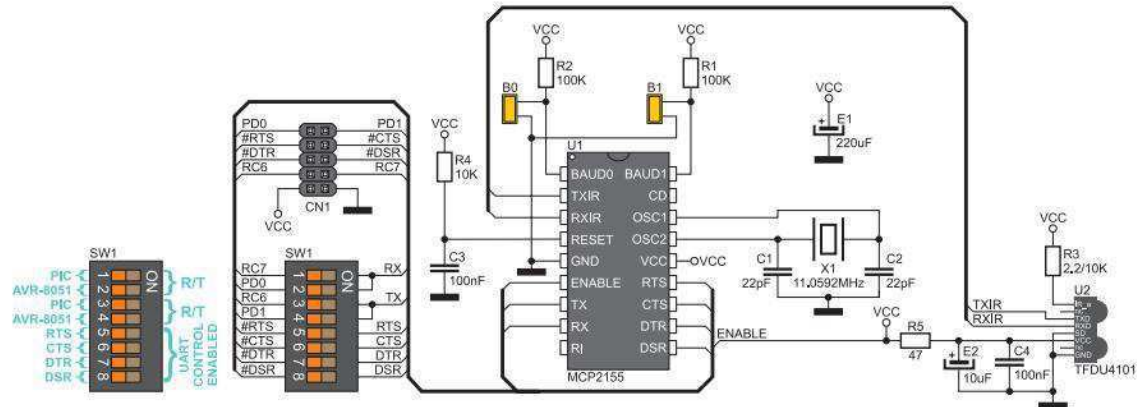


Figura 15 – Diagrama de aplicação IrDA sugerido pela mikroe.

Como módulo transceptor infravermelho para computadores, pode-se utilizar também o dispositivo TFDU 4300 como recomendado na documentação oficial do padrão IrDA, assim como apresentado na figura 16.

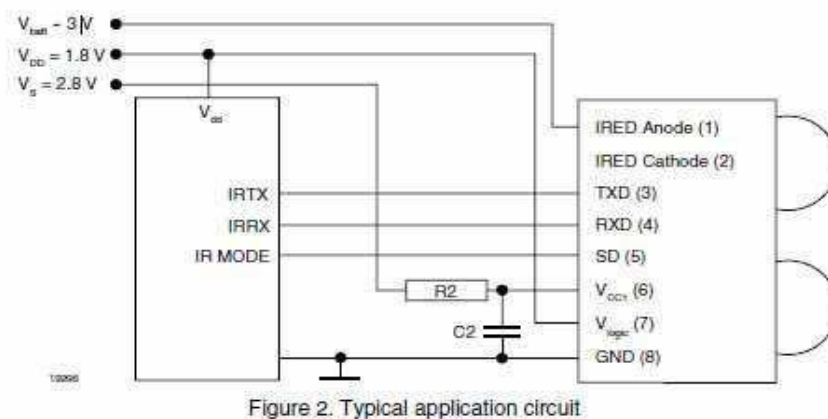


Figure 2. Typical application circuit

Figura 16 – TFDU 4300 (diagrama sugerido na documentação IrDA). [26]

4.2 SIMPLES CONVERSOR IRDA/RS232

Além dos conversores apresentados nos tópicos anteriores, quando se possui uma aplicação mais simples, utilizando apenas a camada física, onde não se faz necessário o estabelecimento de uma conexão como é o caso da comunicação de alguns dispositivos de medição, a utilização de estratégias mais simples para a leitura deste sinal pode ser de grande valia.

É notório o processo de conversão quando se realiza a análise e comparação entre uma palavra transmitida pela IrDA e pela RS232.

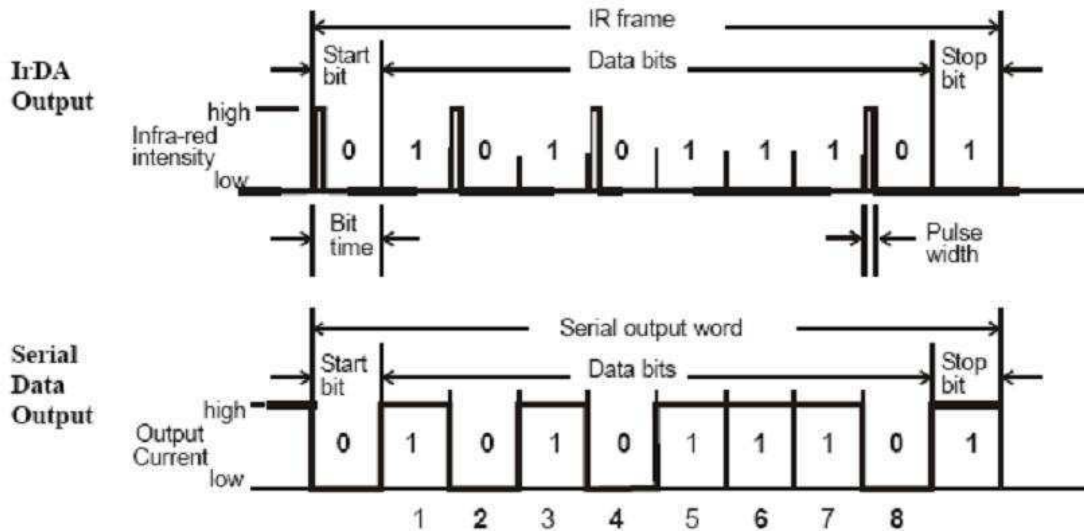


Figura 17 – Comparação no formato da palavra Serial e da IrDA.

Pode-se observar que para converter o sinal IrDA para RS232, basta alongar o pulso positivo correspondente ao campo do bit 0 para o intervalo de 16 pulsos de clock do gerador de oscilação da IrDA segundo o padrão, valor esse associado diretamente a taxa da transmissão, e por fim promover a inversão deste sinal modificado como representado na figura abaixo.

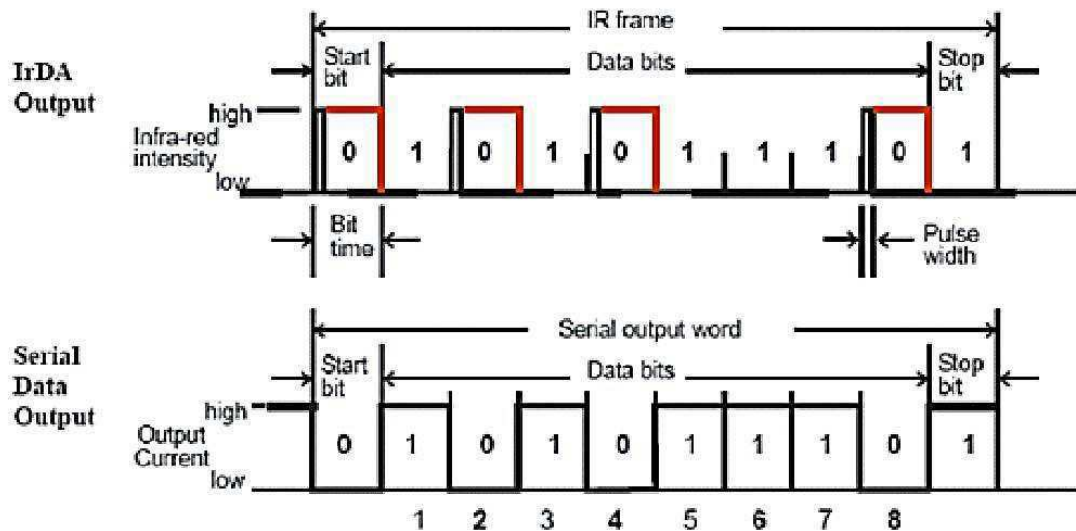


Figura 18 – Processo de conversão IrDA/RS232 (compensação da largura do pulso com temporizador).

Para realizar este procedimento utilizou-se o circuito conhecido como oscilador monoestável com o CI 555 associado a um mecanismo de disparo por meio de um fototransistor, o til 78, cujo esquema do circuito é apresentado na figura 19.

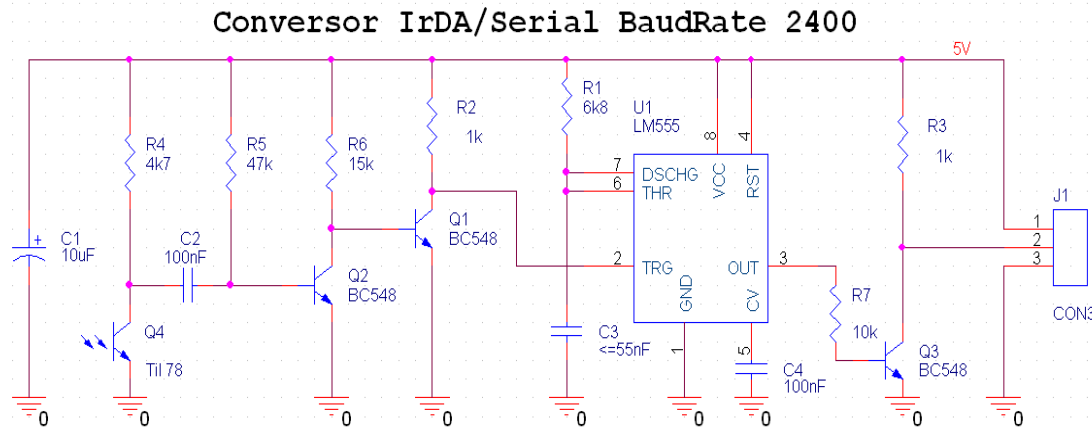


Figura 19 – Esquema do Adaptador do Sinal IrDA para RS232.

O CI 555 é amplamente conhecido no ambiente da eletrônica, possuindo três montagens clássicas, conhecidas como modos monoestável, astável e biestável, possuindo um comportamento de contador, ou temporizador, de oscilador e de flip-flop respectivamente.

No modo monoestável, a tensão de saída permanece em nível baixo enquanto a entrada permanece estável em nível alto e assim que esta recebe uma transição de descida (este momento é conhecido como disparo), faz-se com que a saída comute para nível alto e permaneça neste nível até que o capacitor da montagem carregue sua tensão até $2/3$ do valor da tensão de alimentação do CI, assim como apresentado na figura abaixo.

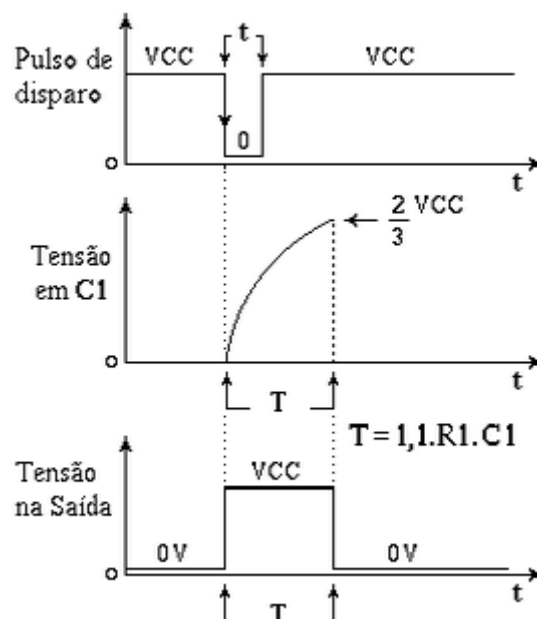


Figura 20 – Análise da resposta do oscilador monoestável 555 ao disparo.

O período que o sinal na saída permanece em nível alto pode ser descrito pela equação de carga do capacitor e corresponde a seguinte equação:

$$T = R1.C1.\ln \frac{V_{cc}}{V_{cc.1-2} 3} \quad (1)$$

$$T = R1.C1.\ln(3) \quad (2)$$

Para cada valor de taxa de comunicação (Baud Rate) tem-se um capacitor diferente para que seja possível a correta adaptação. Abaixo segue uma tabela com alguns valores comuns de baud rate e os respectivos capacitores a serem utilizados quando utilizado um resistor R1 de 6k8.

Tabela 1. Capacitor C1 adotado no Adaptador para um respectivo Baud Rate.

Baud Rate bps	2400	4800	9600
Capacitor C1 (nF)	55,774 (47)	27,887 (27)	13,944 (10)

Para a confecção da placa de circuito impresso realizou-se uma sequência de procedimentos começando pela criação do esquemático do circuito no programa Capture CIS onde é gerado o netlist, importando em seguida este arquivo para o programa Layout Plus onde se realiza o desenho da placa propriamente dita. Tanto o Capture Cis, quanto o Layout Plus fazem parte do software Orcad 9.2. Terminado o desenho da camada Bottom já que a placa só possui uma face, gera-se os arquivos GERBER que são arquivos próprios para máquinas CNC, com suas referências de pads, furos e trilhas.

A máquina CNC por sua vez corta as trilhas e faz a furação restando apenas à soldagem dos componentes para que a placa de circuito impresso esteja pronta assim como apresentado na figura abaixo.



Figura 21 – Montagem do Adptador IrDA/RS232.

4.3 LEITOR IRDA UNIDIRECIONAL COM PIC

Todos os outros circuitos demonstrados ou comentados anteriormente serviam apenas de uma interface entre um computador ou um microcontrolador com dispositivos IrDA. Porém, para que seja possível o estabelecimento de uma comunicação e a utilização das camadas superiores à camada física se faz necessário à implementação de protocolos mostrados anteriormente. Dispositivos como PDAs, Notebooks ou outros dispositivos mais aplicados podem ser custosos quando necessários apenas em pequenas aplicações como realizar a leitura de um medidor de energia e enviar comandos de impressão da conta do usuário para uma impressora térmica. Neste cenário a figura do microcontrolador se apresenta como uma excelente solução, pois além de reduzir custos permite posteriores customizações proporcionando uma melhor flexibilidade aos seus usuários.

Nas aplicações realizadas continuou-se trabalhando apenas com a camada física do sinal da IrDA, porém agora com a adição do microcontrolador é possível realizar testes de comunicação por meio de verificações de valores de Checksum entre outros.

Parte-se agora para experiências de simulação utilizando o software Proteus Isis 7.8 para uma aplicação entre IrDA e o Microcontrolador PIC. Como o dispositivo IrDA não faz parte da biblioteca de componentes do Proteus, utilizou-se um microcontrolador para converter uma mensagem no padrão serial RS232 escrita pelo usuário por meio do componente Virtual Terminal para o padrão IrDA seguindo as especificações da camada física. A saída deste dispositivo por sua vez foi aplicada a uma entrada específica de outro microcontrolador, sendo este responsável por realizar a leitura e coleta dos dados enviados.

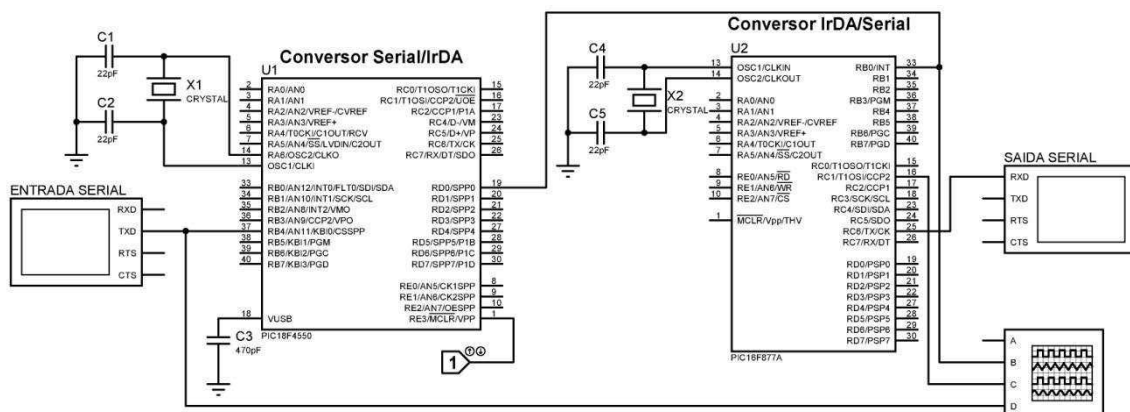


Figura 22 – Simulação Conversor IrDA/Serial.

Pode-se ter uma visão macro do programa do Conversor IRDA/Serial a partir do fluxograma apresentado na figura 23.

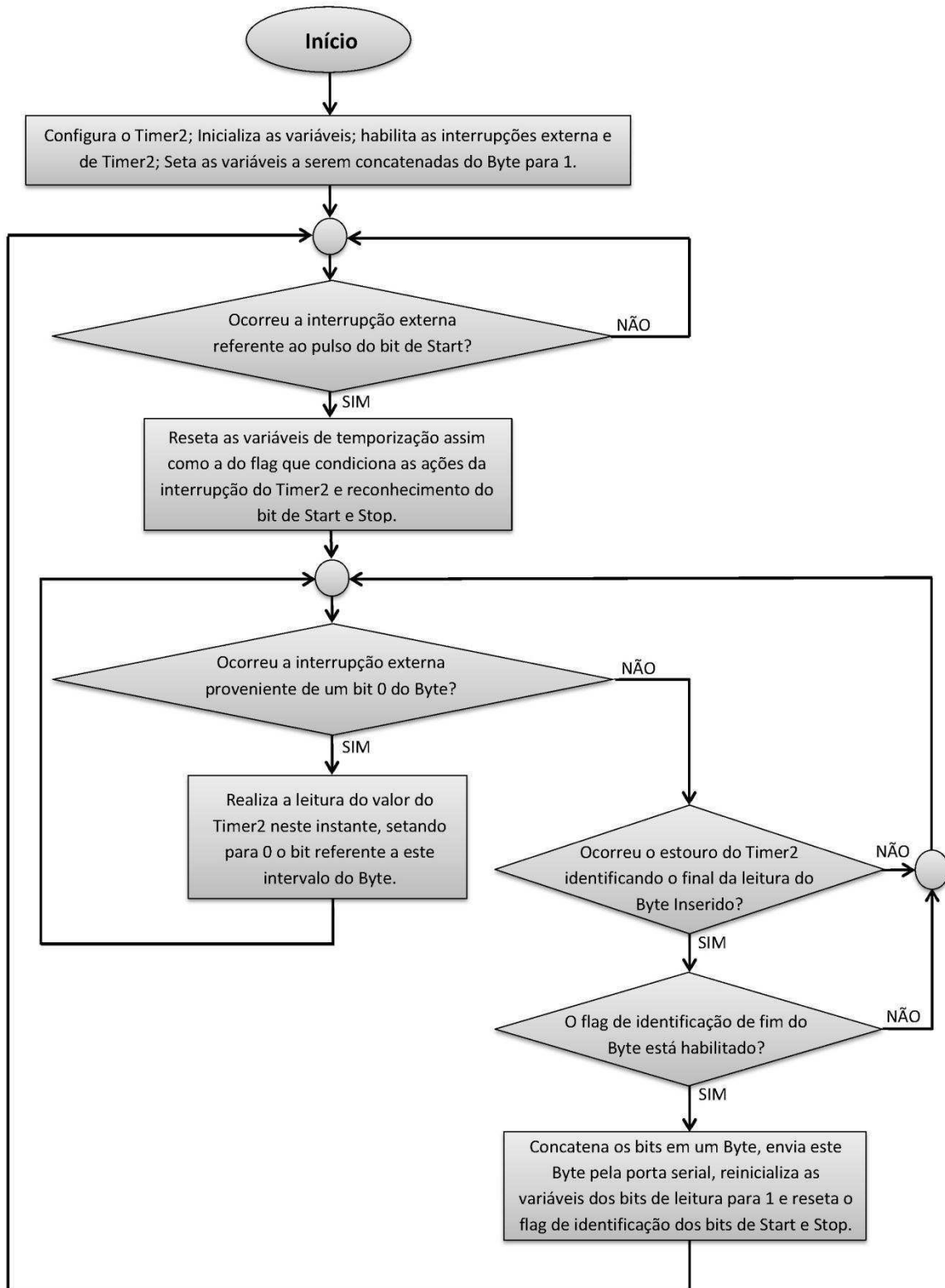


Figura 23 – Fluxograma do programa do Conversor IrDA/Serial.

Para o conversor a estratégia da programação se baseou no cálculo do tempo de permanência do pulso quando enviado um bit 0 além do posicionamento do mesmo no centro do quadro.

Na figura abaixo, demonstra-se a utilização de uma interrupção externa do tipo portB que detecta o instante da transição de descida, momento este caracterizado pelo bit de start.

```

67 #int_RB
68 void RB_isr(void)
69 {
70     pino_atual = input(PIN_B4);
71
72     if(pino_atual > pino_ant);           // subida
73
74     else                                 // descida
75     {
76         cont_byte = 1;
77         disable_interrupts(INT_RB);
78         flag_ini_byte = 1;
79     }
80     pino_ant = pino_atual;
81

```

Figura 24 – Parte do código do conversor Serial/IrDA para a detecção do bit de start.

Logo em seguida no programa principal se tem a adição dos intervalos necessários para a representação do sinal IrDA para um baud rate de 2400bps.

```

92 while(true)
93 {
94     if(flag_ini_byte == 1)
95     {
96         if(cont_byte > 0 && cont_byte <= 9 )
97         {
98             delay_us(169);
99
100            if(cont_byte != 1)
101            {
102                delay_us(169);
103            }
104
105            output_bit(PIN_D0, !input(PIN_B4));
106            delay_us(78);
107            output_low(PIN_D0);
108        }
109
110        if(cont_byte >= 10)
111        {
112            delay_us(416);
113            flag_ini_byte = 0;
114            clear_interrupt(INT_RB);
115            enable_interrupts(INT_RB);
116            pino_ant = input(PIN_B4);
117            cont_byte = 0;
118        }
119        else
120            cont_byte++;
121    }
122

```

Figura 25 – Linhas de código referente a implementação do sinal IrDA.

O código fonte completo desta aplicação se encontra no apêndice e pode-se observar nele os parâmetros de configuração do microcontrolador utilizado, sendo uma das muitas características do PIC a compatibilidade e a utilização de programas para diversos modelos ou famílias.

Para o circuito referente ao conversor IrDA/Serial é interessante observar a forma como são utilizados a interrupção externa em conjunto com o temporizador Timer 2.

A interrupção externa detecta o bit de start sincronizando o temporizador Timer 2 a cada byte recebido o que garante uma excelente precisão na conversão o que não poderia ser obtido por meio de funções de delay como se poderia imaginar inicialmente. Essa deficiência está presente por causa do oscilador adotado para gerar os pulsos de clock para o PIC. Como este não possui um fator multiplicativo inteiro em relação ao baud rate adotado na comunicação, seria impossível uma sincronização perfeita de leitura e conversão.

A interrupção externa sincroniza o temporizador do timer 2 e já a interrupção do timer 2 dita o fim do byte recebido e realiza a preparação ou inicialização das variáveis e registradores para o próximo byte.

```
131 void main()
132 {
133     setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
134     setup_timer_1(T1_DISABLED);
135     setup_timer_2(T2_DIV_BY_16,255,1);
136
137     enable_interrupts(INT_EXT);
138     enable_interrupts(INT_TIMER2);
139     enable_interrupts(GLOBAL);
140
141     printf("CONVERSAO IrDA/SERIAL\n\r");
142
143     while(1);
144 }
```

Figura 26 – Linhas de código relacionadas a configuração do Timer 2.

O principal nesse código está na configuração dos parâmetros do timer 2. Para um baud rate de 2400bps tem-se um período de 4,16 μ s para um quadro completo de 10bits.

Com a utilização de um cristal de 4MHz, um Prescaler (divisor) de 16 e um Postscaler de 1, obtém-se um período de interrupção 4,096 μ s, o que para esta aplicação é o suficiente já que o último bit é o de stop, ocorrendo o tratamento desta interrupção

no meio do intervalo deste bit, fazendo então a concatenação e exibição do byte lido e reinicialização das variáveis necessárias ao processo.

Segue abaixo a equação para a obtenção da frequência e de forma indireta a do período da interrupção: [7]

$$F_{int} = (F_{cristal}/4)/(Prescaler/(256 - PR2))/Postscaler \quad (3)$$

$$F_{int} = 244,14\text{Hz} \quad (4)$$

O resultado da aplicação é apresentado na figura abaixo. As formas de onda amarela, azul, rosa e verde correspondem ao sinal serial aplicado ao conversor Serial/IrDA, ao sinal convertido por este, ao sinal representativo do tratamento e concatenação do byte lido e por fim a representação do sinal de saída do conversor IrDA/Serial respectivamente.

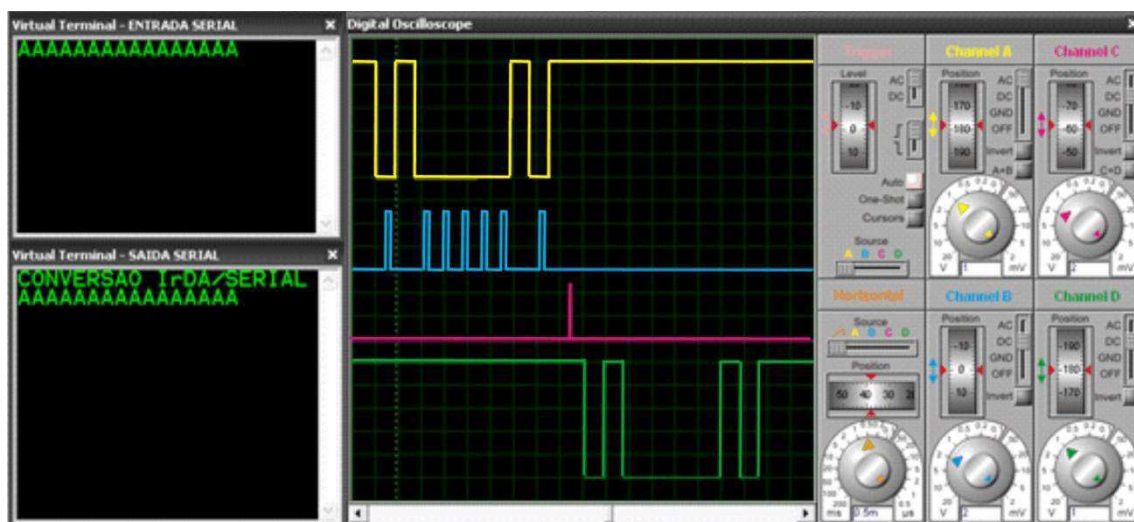


Figura 27 – Análise comparativa entre o byte escrito e o byte lido após processo de conversão.

Ao lado esquerdo estão os Terminais Virtuais representando os caracteres escritos pelo Usuário e os caracteres exibidos ao fim do processo de conversão.

```

104 #int_TIMER2
105 void TIMER2_isr(void)
106 {
107
108     if(flag_estouro_timer == 0)
109     {
110         output_high(PIN_C1);
111         bytelido = (bit7*128)+(bit6*64)+(bit5*32)+(bit4*16)+(bit3*8)+(bit2*4)+(bit1*2)+(bit0);
112         printf("%c", bytelido);
113         mensagem[contador] = bytelido;
114         contador++;
115         flag_estouro_timer = 1;
116         output_low(PIN_C1);
117     }
118
119     bit0 = 1;
120     bit1 = 1;
121     bit2 = 1;
122     bit3 = 1;
123     bit4 = 1;
124     bit5 = 1;
125     bit6 = 1;
126     bit7 = 1;
127

```

Figura 28 – Linhas de Código referente ao tratamento da interrupção do Timer 2.

5 CONCLUSÃO

Se fosse perguntada a qualquer pessoa sobre meios de comunicação por infravermelho para transferência de dados, a primeira impressão seria a de pensar se esta tecnologia não estaria ultrapassada.

Pode-se observar o constante desenvolvimento do padrão IrDA e por meio de melhorias nos protocolos além de novas pesquisas para criação de módulos transceptores mais rápidos conseguiu-se chegar a taxas de transmissão de 1Gbps. Esta taxa mesmo sendo elevada já se trabalha para conseguir marcos de 5 e até 10Gbps.

Por ter esta tecnologia implantada em Smart Meters devido ao baixo custo, cada vez mais, o padrão IrDA vem voltando ao cotidiano.

Microcontroladores são dispositivos excelentes para se trabalhar já que na grande maioria das vezes basta alterar poucas linhas de código para ajustes e adequação do mesmo ao projeto e com as montagens aqui demonstradas ficou clara a força da associação desse dispositivo com a eletrônica.

6 SUGESTÕES PARA TRABALHOS FUTUROS

Como sugestões para trabalhos futuros seria interessante a montagem de um dos circuitos transceptores mostrados na seção 4, conectando-o nos pinos da porta de comunicação por infravermelho podendo assim realizar testes com as outras camadas e protocolos definidos nas especificações.

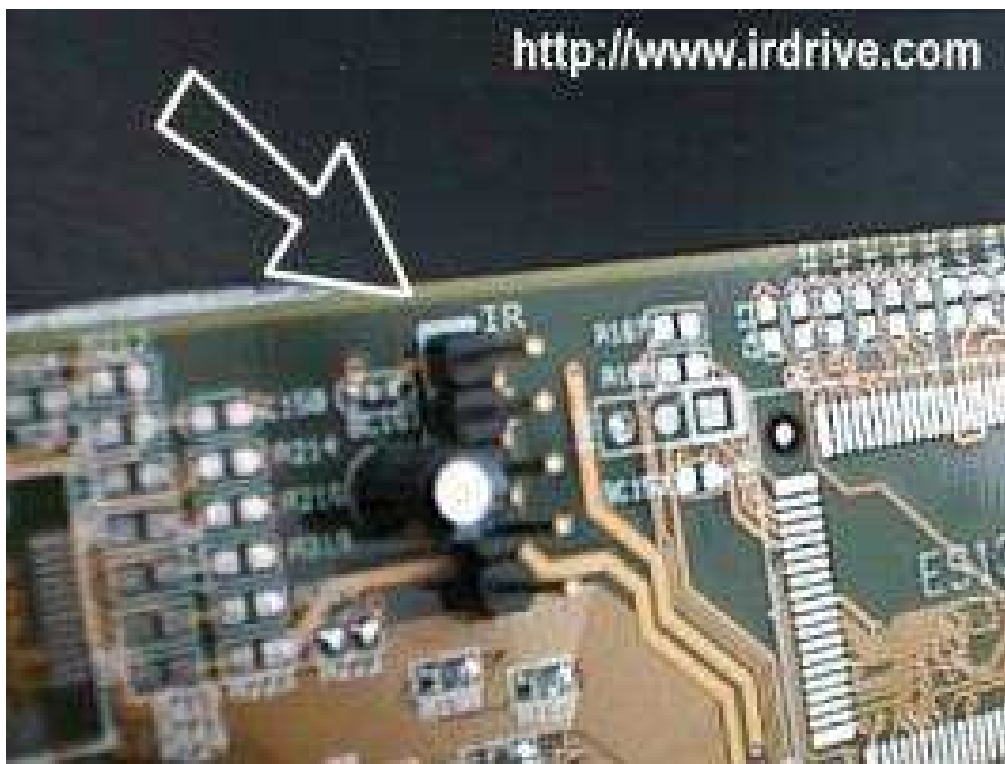


Figura 29 – Pinos da porta IrDA na placa mãe.

Outra opção de montagem seria a de um detector automático de baud rate para interface IrDA com microcontrolador PIC.

BIBLIOGRAFIA

- [1] GAST, M. S. **802.11 Wireless Networks: The Definitive Guide**, Estados Unidos: O'Reilly Media, Inc., 2002, 443p, ISBN 0-596-00183-5.
- [2] JAYANT, N., **Broadband Last Mile: Access Technologies for Multimedia Communications**, Estados Unidos: CRC Press, 2010, 680p, ISBN 0-8247-5886-2.
- [3] JOBSTRAIBIZER, F. **Desvendando as redes sem fio**, São Paulo: Digerati Books, 2010, 96p. ISBN 978-85-7873-102-1.
- [4] NORTON, P.; MUELLER, J. P. **Peter Norton's Complete Guide to Windows XP**, Estados Unidos: Pearson Education, 1997, 752p. ISBN 0672-322-919.
- [5] OSTERHAGE, W. **Wireless Security**, Estados Unidos: CRC Press, 2011, 212p. ISBN 978-1-57808-768-6.
- [6] PEREIRA, F. **Microcontroladores PIC - Programação em C**, São Paulo: Editora Érica, 2003, ISBN 85-7194-935-2.
- [7] PEREIRA, F. **Microcontroladores PIC - Técnicas Avançadas**, São Paulo: Editora Érica, 2002, 360p. ISBN 85-7194-727-9.
- [8] PRASAD, K. V. **Principles of Digital Communication Systems and Computer Networks**, Estados Unidos: Cengage Learning, 2004, 742p. ISBN 1-58450-329-7.
- [9] SARKAR, S. K.; BASAVARAJU, T.G; PUTTAMADAPPA, C. **Ad Hoc Mobile Wireless Networks: Principles, Protocols and Applications**, Estados Unidos: CRC Press, 2007, 336p. ISBN 978-1-4200-6221-2.
- [10] SOUZA, D. J. **Desbravando o PIC: ampliado e atualizado para PIC16F628A**, 8ª edição, São Paulo: Érica, 2005, 268p. ISBN 85-7194-867-4.

[11] Site oficial IrDA, Disponível em <<http://www.irda.org>>, Acessado em 26 set. 2013.

[12] Lista de Membros IrDA, Disponível em <<http://irdajp.info/memberslist.html>>, Acessado em 26 set. 2013.

[13] Padrão IrDA, Disponível em <<http://irdajp.info/specifications.html>>, Acessado em 26 set. 2013.

[14] IrDA Data Link Design Guide, Disponível em <<http://www.datasheetarchive.com/dlmain/Datasheets-17/DSA-332168.pdf>>, Acessado em 28 set. 2013

[15] Agilent IrDA Data Link Design Guide, Disponível em <http://www.rdc.cz/prilohy/hardware/53_IrDA%20popis%20AGILENT%20tech.pdf>, Acessado em 28 set. 2013

[16] Raios infravermelhos, Disponível em <<http://www.brasilecola.com/quimica/raios-infravermelhos.htm>>, Acessado em 22 set. 2013

[17] Guia do Hardware – IrDA, Disponível em <<http://www.hardware.com.br/termos/irda>>, Acessado em 23 set. 2013

[18] Engenharia de Domínio, Disponível em <<http://www.lisha.ufsc.br/~guto/teaching/os2/ine5357-2002-1/work/irda/index.html>>, Acessado em 30 set. 2013

[19] IrDA Board, Disponível em <<http://www.mikroe.com/add-on-boards/communication/irda/#>>, Acessado em 23 set 2013

[20] IrDACompatible Data Transmission/IrDA–Standard – Physical Layer, Disponível em <http://www.ele.uva.es/~jesman/BigSeti/ftp/Comunicaciones/IrDA/irdc_physical_layer.pdf>,Acessado em 25 set. 2013

[21] Infrared Data Communications with IrDA , Disponível em: <http://electronica.ugr.es/~amroldan/modulos/temas_tecnicos/irda/infrared_data_communications_with_irda.pdf> Acessado em 25 set. 2013

[22] Wearable remote control, Disponível em : <http://www.d-sciencelab.be/i-ware/dsp_case1aed.html?case_id=345> , Acessado em 23 set 2013

[23] IrDa e o AT89C51, Disponível em: <http://ltodi.est.ips.pt/microcont/contribuicaoalunos/IrDa/IrDa_micro.htm.> Acessado em: 23 set 2013

[24] Circuit Enables PCs to Communicate via IrDA, Disponível em: <<http://www.maximintegrated.com/app-notes/index.mvp/id/631>>, Acessado em 23 set. 2013

[25] IrDA transceiver for PC, Disponível em: <http://pavouk.org/hw/en_irda.html>, Acessado em: 30 set. 2013

[26] IrDA Enabling a PC or Microcontroller, Disponível em <http://trandi.wordpress.com/2010/01/09/irda_mcp2120_tfd4300_ftdi_ttl-232_capacitor_crystal_7-3728/>, Acessado em 23 set 2013

APÊNDICE A – PROGRAMA DO CONVERSOR

IRDA/SERIAL (SIMULAÇÃO NO PROTEUS)

/*****
 Universidade Federal de Campina Grande - UFCG / Projeto de Conclusão de Curso

Nome do Programa: conv_irda_serial_timer_04.c

local armazenado:

C:\Documents and Settings\Ramses\Desktop\leitor de dados\conversor serial_irda

Autor: Ramsés Araujo Gonçalves

Descrição: Programa que converte o sinal IRDA em serial (RS232) 2400bps.

Características de Hardware:

*Conecta-se a saída da IRDA (nível TTL) na entrada do Pino RB0 e tem-se a saída serial RS232(TTL) correspondente a conversão no pino TX (RC6).

*****/

```
#include <16F877A.h>
```

```
#device adc=8
```

```

#FUSES NOWDT           //No Watch Dog Timer
#FUSES HS              //High speed Osc (> 4mhz for PCM/PCH) (>10mhz for PCD)
#FUSES PUT            //Power Up Timer
#FUSES NOPROTECT      //Code not protected from reading
#FUSES NODEBUG        //No Debug mode for ICD
#FUSES NOBROWNOUT     //No brownout reset
#FUSES NOLVP          //No low voltage prgming, B3(PIC16) or B5(PIC18) used for I/O
#FUSES NOCPD          //No EE protection
#FUSES NOWRT          //Program memory not write protected

```

```
#use delay(clock=4000000)
```

```
#use rs232(baud=2400,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8)
```

```
int1 flag_estouro_timer = 1;
```

```
int flag_imprima = 6;
```

```
unsigned int timer = 0;
```

```
unsigned int bytelido = 0;
```

```
int contador = 0;
```

```
char mensagem[90] = "\0";
```

```
int bit0 = 1;
```

```
int bit1 = 1;
```

```
int bit2 = 1;
```

```
int bit3 = 1;
int bit4 = 1;
int bit5 = 1;
int bit6 = 1;
int bit7 = 1;

#int_EXT
void EXT_isr(void)
{
    if(flag_estouro_timer)
    {
        set_timer2(13);
        timer = 0;
        flag_estouro_timer = 0;
    }
    else
    {
        timer = get_timer2();
        if(timer > 24 && timer <= 52)
        {
            bit0 = 0;
        }
        if(timer > 52 && timer <= 76)
        {
            bit1 = 0;
        }
        if(timer > 76 && timer <= 104)
        {
            bit2 = 0;
        }
        if(timer > 104 && timer <= 128)
        {
            bit3 = 0;
        }
        if(timer > 128 && timer <= 156)
        {
            bit4 = 0;
        }
        if(timer > 156 && timer <= 180)
        {
            bit5 = 0;
        }
        if(timer > 180 && timer <= 208)
        {
            bit6 = 0;
        }
        if(timer > 208 && timer <= 232)
        {
            bit7 = 0;
        }
        if(timer > 232 )
    }
}
```



```

    {
    }
    output_low(PIN_C0);
}
}

#int_TIMER2
void TIMER2_isr(void)
{
    if(flag_estouro_timer == 0)
    {
        output_high(PIN_C1);
        bytelido = (bit7*128)+(bit6*64)+(bit5*32)+(bit4*16)+(bit3*8)+(bit2*4)+(bit1*2)+(bit0);
        printf("%c", bytelido);
        mensagem[contador] = bytelido;
        contador++;
        flag_estouro_timer = 1;
        output_low(PIN_C1);
    }

    bit0 = 1;
    bit1 = 1;
    bit2 = 1;
    bit3 = 1;
    bit4 = 1;
    bit5 = 1;
    bit6 = 1;
    bit7 = 1;
}

void main()
{
    setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
    setup_timer_1(T1_DISABLED);
    setup_timer_2(T2_DIV_BY_16,255,1);

    enable_interrupts(INT_EXT);
    enable_interrupts(INT_TIMER2);
    enable_interrupts(GLOBAL);

    printf("CONVERSAO IrDA/SERIAL\n\r");

    while(1);
}

```

APÊNDICE B – PROGRAMA DO CONVERSOR

SERIAL/IRDA (SIMULAÇÃO NO PROTEUS)

/*****
 Universidade Federal de Campina Grande - UFCG / Projeto de Conclusão de Curso

Nome do Programa: conv_serial_irda_teste6.c

local armazenado:

C:\Documents and Settings\Ramses\Desktop\leitor de dados\conversor serial_irda

Autor: Ramsés Araujo Gonçalves

Descrição: Programa que converte o sinal da porta serial (RS232) em IRDA.

Características de Hardware:

*Conecta-se a saída TX da serial (nível TTL) na entrada do Pino RB4 e tem-se a saída IRDA no Pino RD0.

*****/

```
#include <18F4550.h>
```

```
#device adc=8
```

```

#FUSES NOWDT           //No Watch Dog Timer
#FUSES WDT128         //Watch Dog Timer uses 1:128 Postscale
#FUSES HS             //High speed Osc (> 4mhz for PCM/PCH) (>10mhz for PCD)
#FUSES NOPROTECT     //Code not protected from reading
#FUSES NOBROWNOUT    //No brownout reset
#FUSES BORV20        //Brownout reset at 2.0V
#FUSES PUT           //Power Up Timer
#FUSES NOCPD         //No EE protection
#FUSES STVREN        //Stack full/underflow will cause reset
#FUSES NODEBUG       //No Debug mode for ICD
#FUSES NOLVP         //No low voltage prgming, B3(PIC16) or B5(PIC18) used for I/O
#FUSES NOWRT         //Program memory not write protected
#FUSES NOWRTD        //Data EEPROM not write protected
#FUSES IESO          //Internal External Switch Over mode enabled
#FUSES FCMEN         //Fail-safe clock monitor enabled
#FUSES PBADEN        //PORTB pins are configured as analog input channels on RESET
#FUSES NOWRTC        //configuration not registers write protected
#FUSES NOWRTB        //Boot block not write protected
#FUSES NOEBTR        //Memory not protected from table reads
#FUSES NOEBTRB       //Boot block not protected from table reads
#FUSES NOCPB         //No Boot Block code protection
#FUSES MCLR          //Master Clear pin enabled
#FUSES LPT1OSC        //Timer1 configured for low-power operation
#FUSES NOXINST       //Ext. set Addressing mode disabled (Legacy mode)

```

```

#FUSES PLL12           //Divide By 12(48MHz oscillator input)
#FUSES CPUDIV4        //System Clock by 4
#FUSES USBDIV         //USB clock source comes from PLL divide by 2
#FUSES VREGEN         //USB voltage regulator enabled
#FUSES ICPRT          //ICPRT enabled

#include <stdlib.h>
#include <string.h>

#use delay(clock=2000000)
#use rs232(baud=2400,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8)

int pino_atual = 0;
int pino_ant = 1;
unsigned int cont_byte = 0;
int flag_ini_byte = 0;

#int_RB
void RB_isr(void)
{
    pino_atual = input(PIN_B4);

    if(pino_atual>pino_ant);           // subida

    else                               // descida
    {
        cont_byte = 1;
        disable_interrupts(INT_RB);
        flag_ini_byte = 1;
    }
    pino_ant = pino_atual;
}

void main()
{
    enable_interrupts(INT_RB);
    enable_interrupts(GLOBAL);

    pino_ant = input(PIN_B4);

    while(true)
    {
        if(flag_ini_byte == 1)
        {
            if(cont_byte > 0 && cont_byte <= 9 )
            {
                delay_us(169);

                if(cont_byte != 1)
                {

```

```
    delay_us(169);
  }

  output_bit(PIN_D0,!input(PIN_B4));
  delay_us(78);
  output_low(PIN_D0);
}

if(cont_byte >= 10)
{
  delay_us(416);
  flag_ini_byte = 0;
  clear_interrupt(INT_RB);
  enable_interrupts(INT_RB);
  pino_ant = input(PIN_B4);
  cont_byte = 0;
}
else
  cont_byte++;
}
}
}
```