

GERENCIAMENTO REMOTO DE APARELHOS ELETRÔNICOS UTILIZANDO PDAs E BLUETOOTH

J. H. McComb

Diretoria de Pesquisa e Pós-Graduação – DIPESP – CEFET-AM
Av. Sete de Setembro, 1973 Centro CEP 69.020-120 Manaus-AM
E-mail: dipesp@cefetam.edu.br

V. F. Lucena Jr.

Diretoria de Pesquisa e Pós-Graduação – DIPESP – CEFET-AM
Av. Sete de Setembro, 1973 Centro CEP 69.020-120 Manaus-AM
E-mail: lucena@cefetam.edu.br

RESUMO

No que concerne à tecnologia móvel, os sistemas de informática têm evoluído bastante. *Handheld Computers* ou PDAs (*Personal Digital Assistant*), como são chamados os computadores de mão, antes pouco atraentes por sua baixa capacidade computacional, hoje agregam recursos mais sofisticados e novos padrões de comunicação, consolidando-se cada vez mais como verdadeiros computadores portáteis. Métodos modernos de telecomunicações permitem que estes dispositivos se conectem a outros com maior poder de processamento ou a dispositivos domésticos gerando as chamadas redes pessoais. Um exemplo dessas novas tecnologias é a denominada *Bluetooth*, padrão e meio de comunicação sem fios integrado não apenas aos PDAs modernos, mas também encontrado em: computadores portáteis, telefones celulares, *smartphones* (telefones inteligentes) e outros equipamentos eletrônicos, inclusive em eletrodomésticos. O trabalho apresentado neste artigo, descreve inicialmente a arquitetura dos PDAs e peculiaridades quanto ao desenvolvimento de aplicações para estes dispositivos. Especial atenção será dada ao uso da tecnologia *Bluetooth* para executar a comunicação com outros dispositivos. Em um segundo momento é apresentado um estudo de caso direcionado ao gerenciamento remoto de aparelhos eletrônicos. Foram desenvolvidas duas aplicações, uma do lado do PC e outra do lado do PDA, as quais interpretam as informações enviadas por esses dispositivos em ambos os lados. Ao sistema do lado do PC caberá o envio de sinais digitais a uma das portas de entrada/saída do computador a fim de controlar o *hardware* responsável pela ativação/desativação dos equipamentos eletrônicos a ele conectados. Um PDA equipado com dispositivo *Bluetooth* será utilizado para controlar os equipamentos à distância. Como o controle também poderá ser realizado diretamente através do PC, regras para manter as informações em ambos os lados sempre atualizadas deverão ser estabelecidas. No trabalho aqui descrito foi implementado, hardware e software no CEFET-AM, representando inclusive nossa instituição em uma Feira de Tecnologia realizada em Porto Alegre.

PALAVRAS-CHAVE: Comunicação sem Fios; Bluetooth; Gerenciamento Remoto; Desenvolvimento para Sistemas Embarcados; Personal Digital Assistants (PDAs); Redes Pessoais.

1. INTRODUÇÃO

A construção de computadores e de outros dispositivos ligados à ciência da computação cresce em desenvolvimento tecnológico e na sua inserção no mundo moderno. Ter computador em casa resolvendo tarefas rotineiras do dia-a-dia não é mais novidade em países do chamado primeiro mundo, paralelamente estes equipamentos vêm se popularizando cada vez mais nos países emergentes. De fato, temos hoje em dia computadores espalhados por toda parte. Isto se dá pelo fato da indústria de computadores sempre perseguir a construção de artefatos, por um lado mais velozes, com maior capacidade de memória, com maior poder de processamento e cada vez menores, e por outro lado sempre mais baratos e acessíveis a mercados maiores.

O pequeno tamanho atingido para muitos destes novos dispositivos computacionais permitiu criar-se uma abrangência de aplicações inimagináveis anteriormente. O que até pouco tempo era feito em grandes computadores de mesa, pode ser feito nos dias atuais em computadores móveis, compactos e portáteis, chegando até o tamanho de uma palma de mão. Os *palmtops*, *handheld computers* ou PDAs (*Personal Digital Assistant*), comumente chamados de computadores de mão, antes pouco atraentes por sua baixa capacidade computacional, atualmente agregam recursos sofisticados ampliando a sua possibilidade de uso. Além disso, seu antes alto custo de aquisição não é mais uma realidade o que o torna acessível a uma quantidade maior de clientes. Tudo isso tem aproximado essas máquinas cada vez mais do cotidiano de grande parte da população e levado diversas empresas a pensar em formas de utilizá-las em seu negócio.

Juntamente com o avanço tecnológico de computadores observamos um grande avanço em técnicas de telecomunicações e de comunicação de dados. Através destas tecnologias vários computadores e dispositivos afins conseguem conversar entre si, aumentando sua usabilidade através da inserção de novas aplicações para facilitar ainda mais a vida dos usuários.

De fato a telecomunicação é um fator importante para a proliferação de dispositivos de mão e tem contribuído para sua evolução, incorporando aos assistentes pessoais dispositivos de comunicação sem fio (*wireless*) cada vez menores. Um exemplo disso são os dispositivos baseados na tecnologia *Bluetooth*, padrão de comunicação integrado não apenas em computadores de mão (PDAs) como também em telefones celulares, *smartphones* (telefones inteligentes), fones de ouvido e outros equipamentos eletrônicos.

No entanto, novos obstáculos são enfrentados ao se desenvolver aplicações para equipamentos móveis e que utilizam esse tipo de comunicação. Dentre os quais podemos citar os limitados recursos disponíveis nesses equipamentos e menor confiabilidade no tráfego de informações se comparadas às redes que utilizam fiação. Além disso, enquanto o desenvolvimento de software para computadores de mesa é uma matéria dominada com técnicas e métodos consolidados na academia e na indústria, o desenvolvimento de software para sistemas menores como os PDAs ainda carece de investigação teórica e prática.

O projeto apresentado neste trabalho investiga esta importante área da informática. Um PDA foi utilizado para controlar um computador à distância que por sua vez atua como controlador de uma série de aparelhos domésticos. Para isso foi necessário desenvolver um software para o PDA com a capacidade de gerenciar o controle desejado dos aparelhos eletrônicos. Este PDA comunica-se com o computador através da tecnologia de telecomunicações sem fios *Bluetooth*. O computador de mesa por sua vez, possui outro software capaz de receber e entender os comandos oriundos do PDA. Utilizando uma de suas portas físicas de comunicação o computador aciona um dispositivo de hardware que permite ligar ou desligar os equipamentos domésticos. O dispositivo de hardware utilizado neste projeto foi desenvolvido especialmente para esta aplicação e usou componentes eletrônicos convencionais, encontrados em qualquer fornecedor de peças. O custo obtido foi muito atrativo tornando o projeto viável do ponto de vista técnico e econômico. O sistema hardware, software do PDA, software do computador de mesa foi integrado e gerou um produto interessante tecnicamente e muito prático para usuários em geral.

As inovações científicas e tecnológicas presentes neste trabalho serão apresentadas como comunicação científica (artigo científico) pela primeira vez neste I CONNEPI a ser realizado em Natal em 2006. No entanto, vale destacar que este projeto foi apresentado em diversas feiras tecnológicas em Manaus (Semana de Ciência e Tecnologia do Estado do Amazonas, Feira Tecnológica da Universidade Federal do Amazonas, etc) e no sul do país. Obteve primeiro lugar na EXPOTEC do CEFET-AM em 2004 e foi convidado pela SETEC do MEC para representar os CEFET na feira de inovação tecnológica Globaltech, realizada em Porto Alegre no ano de 2005.

Na sequência serão apresentados os objetivos originais deste projeto com suas justificativas na seção 2. Na seção 3, são descritos os materiais e os métodos utilizados para a execução deste trabalho. Na seção 4 será detalhado o processo de desenvolvimento tanto do software quanto do hardware utilizado neste projeto. Os resultados obtidos são comentados na seção 5. Na seção 6 são registradas as conclusões obtidas e finalmente na seção 7 são apresentadas as referências mais importantes.

2. OBJETIVOS

O sistema de gerenciamento remoto teve como principal objetivo efetuar o controle à distância, com o auxílio de equipamentos portáteis (PDAs) e meios de comunicação sem fio (*Bluetooth*), de diversos equipamentos eletrônicos, tais como: lâmpadas, portas elétricas, motores, etc. São necessários então um PDA e um computador pessoal (PC), que assumimos ser cada vez mais popular e disponível em vários lares e um dispositivo de hardware construído especificamente para garantir que os comandos recebidos pelo computador sejam de fato executados.

Para atingir este objetivo foi necessário realizar uma série de estudos e pesquisas que envolveram:

- a) Estudar a arquitetura dos PDAs e estudar técnicas de desenvolvimento de software para estes dispositivos;
- b) Estudar o padrão de comunicação sem fio *Bluetooth*;
- c) Projetar um protocolo de comunicação de dados baseado em *Bluetooth*;
- d) Desenvolver um esboço da arquitetura do sistema a ser construído;
- e) Construir o protótipo de um equipamento digital de x entradas e n saídas capazes de acionar aparelhos eletrônicos, sendo n maior que x ;
- f) Desenvolver uma ferramenta de *software* para o PC capaz de controlar esse equipamento através da porta paralela e que possa ser controlado via *Bluetooth*;
- g) Desenvolver um *software* para o PDA que se comunique com o PC através dessa tecnologia *wireless*;
- h) Implementar regras de comunicação de modo a manter o controle do fluxo de dados e assegurar o sincronismo em tempo real das informações contidas tanto no PC quanto no PDA sobre os aparelhos conectados ao sistema;
- i) Integrar o sistema e testá-lo com dispositivos reais.

2.1. Justificativas

A troca de informações entre equipamentos eletrônicos antes incomunicáveis é considerada uma tendência cada vez mais popular e barata. Este projeto é a aplicação de um importante padrão de comunicação *wireless* utilizado para este fim, o *Bluetooth*, vindo a colaborar para o enriquecimento do conhecimento científico e oferecendo mais uma opção de integração do homem às tecnologias de ponta.

Uma opção analisada foi o uso de infra-vermelho como meio de comunicação sem fios entre o PDA e o computador. No entanto a característica de alcance e de visada desta tecnologia a tornou menos atrativa. Infra-vermelho tem um alcance inferior a *Bluetooth* e precisa ter uma visada direta entre os dois pontos comunicantes para garantir a troca de dados, *Bluetooth* é omnidirecional irradiando suas ondas de comunicação para todos os lados tendo o dispositivo de mão como centro irradiador. Além disso, observa-se que existem cada vez mais equipamentos que possuam este meio de comunicação. Até mesmo telefones celulares de baixo custo são capazes de comunicar-se entre si ou com computadores usando esta tecnologia.

A idéia base se consolidou com a necessidade pessoal de integrar a um baixo custo o controle de equipamentos domésticos, valendo-se daquilo que há de mais moderno em tecnologia. O fato é que com computadores de mesa e PDAs cada vez mais presentes na vida moderna certas funcionalidades podem ser implementadas e desenvolvidas através de software. O software desenvolvido permitiu criar um controle remoto a ser executado no PDA capaz de controlar diversos dispositivos eletrônicos.

Contudo, visto que os aparelhos eletrônicos gerenciados pelo sistema não possuem um padrão de comunicação sem fios em comum, fez-se necessária a elaboração de um equipamento que os intermediasse, servindo de *interface* entre o PC e todos os aparelhos externos ao sistema. Este equipamento é responsável por acionar relés que garantam a conexão dos equipamentos domésticos à fonte de energia elétrica que garanta que os mesmos sejam ligados.

A centralização das tarefas transfere comodidade ao usuário, permitindo seu controle de qualquer ponto da residência ou escritório. Mas isso só é possível pela utilização dos dispositivos *wireless* dos PDAs que, pelas suas características, apresenta-se como grande atrativo a um projeto que pode assumir também um fim comercial e ser aplicado, principalmente, no ramo doméstico.

3. DESCRIÇÃO DO SISTEMA DESENVOLVIDO

Como já foi mencionado, o sistema de gerenciamento remoto foi dividido em duas partes principais, uma parte lógica, chamada de gerenciador remoto, e uma parte física, o controlador digital.

O gerenciador remoto por sua vez não se trata apenas de um programa de computador; na verdade, são duas aplicações desenvolvidas para plataformas diferentes que mantêm informações semelhantes e as sincronizam em tempo real. O gerenciamento se dá por meio dessas aplicações que atuam de forma conjunta, sendo que uma funciona do lado do PDA e a outra do lado do PC. Outra função do gerenciador remoto é o controle e visualização dos estados dos diversos aparelhos eletrônicos acoplados ao sistema. Basicamente, a aplicação do PDA envia sinais de controle para o sistema do PC e o sistema do PC interpretará esses sinais, acionando o hardware a ele conectado. A comunicação é sem fio. A Figura 1 dá uma visão geral desse esquema.

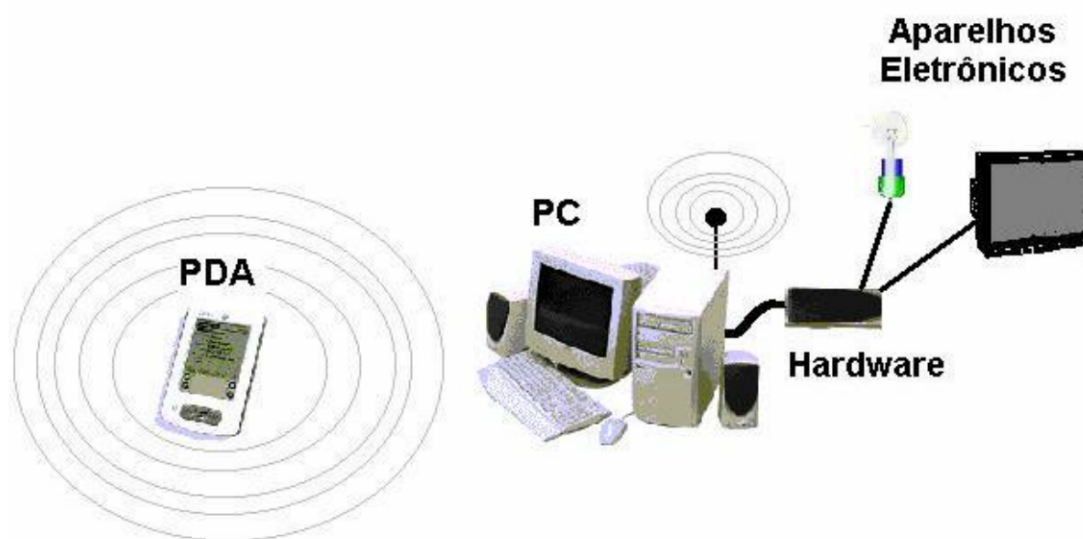


Figura 1. Esquema geral do sistema de gerenciamento remoto.

Por fim, o controlador digital, *hardware* conectado diretamente à porta paralela do PC, é a *interface* entre o computador e os aparelhos externos, cabendo-lhe a função de acioná-los eletronicamente. Este controlador recebe informações sobre qual equipamento deve ser acionado ou desligado e leva a efeito esta ação. O aparelho externo fica neste estado até que venha um outro comando alterando-o.

O estudo da arquitetura dos PDAs e dos padrões de comunicação atualmente existentes fez-se necessário, a fim escolher aquele mais adequado ao projeto. Alves (2002) faz uma comparação entre as duas plataformas mais usadas em PDAs, Palm OS e o Windows CE. Muito mais do que o sistema operacional utilizado, a escolha do PDA mais adequado deve levar em conta as ferramentas de desenvolvimento disponíveis, as linguagens de programação possíveis de serem empregadas, memória existente, etc. Após tal análise, decidiu-se neste projeto pela adoção do computador de mão *Palm Zire 72*, com 32MB de memória RAM, *Bluetooth* embutido e sistema operacional Palm OS 5.

O computador deve ser um PC que suporte Windows. O sistema demanda pouco processamento e pouca memória o que permite que seja instalado em um grande número de computadores já disponíveis nos lares de potenciais usuários. Desejável seria que o PC viesse com *Bluetooth* embutido permitindo uma comunicação direta entre ele e o PDA. No entanto, foi possível reduzir esta exigência utilizando um adaptador *Bluetooth*. Desta forma, a configuração mínima para o PC passou a exigir uma entrada USB, a fim de que fosse conectado a ela um adaptador *Bluetooth* USB, da Actiontec, modelo BTM200B. Deve ficar claro que o sistema foi pensado para que outros modelos de PDAs dotados com a mesma plataforma (Palm OS) e outros modelos de adaptadores *Bluetooth* para PCs possam vir a ser utilizados como plataforma base.

Nas seções a seguir descreve-se o trabalho desenvolvido na construção deste sistema. Primeiramente, será abordada a metodologia utilizada para o desenvolvimento do gerenciador remoto (*software*), logo após serão vistos detalhes sobre o controlador digital (*hardware*).

3.1. Gerenciador Remoto

O gerenciador remoto é um software desenvolvido usando conceitos de orientação a objetos associados à metodologia ICONIX. A abordagem orientada a objetos procura diminuir a distância entre os conceitos da aplicação no mundo real e a implementação através de uma linguagem de programação. O mesmo conceito de objeto é usado ao longo de todas as fases do desenvolvimento em diferentes níveis de abstração.

Muitas metodologias de desenvolvimento de software orientadas a objetos surgiram no início da década de 80. Na ocasião estas metodologias foram apresentadas como o grande Eldorado da ciência da computação e que resolveriam todos os problemas de desenvolvedores e analistas. Hoje, orientação a objetos tem seu lugar definido na história da computação e ainda aparece como uma abordagem sólida para desenvolvimento e de fácil acesso, ver Sintès (2002). Assim, baseadas na noção de metodologias unificadas, destacando-se as seguintes:

- a) RUP (*Rational Unified Process*), mais teórica, abrangente e completa (ver Kruchten (1999));
- b) XP (*Extreme Programming*) apropriada para pequenos projetos (ver Ambler e Jeffries (2002));
- c) ICONIX, pura, prática e simples, mas também com um componente de análise e representação dos problemas sólido e eficaz. (Bona e Costa (2004))

Das três metodologias citadas, adotou-se o ICONIX para o desenvolvimento do sistema. Metodologia não muito simples quanto o XP e nem muito complexa quanto o RUP, porém, poderosa. O ICONIX é uma síntese obtida a partir dos métodos originais de Booch, Rumbaugh e Jacobson, após 10 anos de refinamentos. Ver Rumbaugh et al (1998)

O ICONIX é orientado a objetos e utiliza o UML (*Unified Modelling Language*) como linguagem de modelagem, retirando um conjunto mínimo de métodos, no entanto, o suficiente para a documentação do sistema. Além disso, o ICONIX é um processo de desenvolvimento de *software* dirigido por casos de uso, iterativo e incremental, desenvolvido pela *Iconix Software Engineering*.

Este trabalho seguiu as quatro fases básicas do ICONIX, que são:

- a) Análise de requisitos;
- b) Análise e projeto preliminar;
- c) Projeto;
- d) Implementação.

Segundo Bona e Costa (2004), “na fase de análise de requisitos são analisadas as idéias base que definem o sistema, procedendo à identificação dos objetos e relações que caracterizam de forma completa e inequívoca do universo a ser representado”. Trata-se de uma fase muito importante que só terá sucesso se existir uma comunicação efetiva entre o cliente que está demandando o trabalho e a equipe desenvolvedora. Segundo Leffingwell e Widrig (2000) é fundamental compreender o que se deseja e mostrar isso ao cliente para garantir que não existam mal entendidos.

Na fase de análise e projeto preliminar são realizadas atividades de análise e construção de diagramas, descrevendo os casos de uso com os cenários principais e alternativos. O objetivo é descrever o sistema a ser desenvolvido de forma que qualquer projetista possa dar continuidade ao trabalho. Toda e qualquer decisão aqui tomada deve fazer referência aos itens levantados na análise de requisitos por outro lado não se deve inserir nada que não tenha sido solicitado pelos clientes na sua definição de requisitos.

A fase de projeto especifica o comportamento do sistema, identificando os objetos e caracterizando as mensagens trocadas entre eles e os métodos associados que serão invocados quando das ações que chamam estas mensagens. O resultado desta fase serão os chamados diagramas de arquitetura de alto nível e de baixo nível contendo descrições detalhadas das funcionalidades dos objetos existentes. Define-se ainda, completamente, os tipos das mensagens e dos métodos associados a cada objeto projetado. Esta etapa é imediatamente anterior à implementação do sistema.

Finalmente, na fase de implementação se tem a correta tradução das atividades anteriores num produto final concreto e estruturado, versado para as funcionalidades pedidas, pelo estudo dos diagramas obtidos a partir da análise de requisitos. Somente aqui se decide por qual linguagem de programação a ser utilizada. É nesta fase que são realizados os testes funcionais e a validação do sistema.

Cada uma das fases de desenvolvimento acima descritas possui diversas técnicas particulares que permitem a sua completitude. O uso destas técnicas depende de quão detalhado deve ser o desenvolvimento do sistema. Neste projeto foram adotadas as seguintes técnicas:

- a) Análise de requisitos
 - Construção de protótipos de interfaces;
 - Desenho de alto nível dos casos de uso.
- b) Análise e projeto preliminar
 - Desenho dos casos de uso com os cenários principais, alternativos e de exceção.
- c) Projeto
 - Desenho de diagramas de seqüência;
 - Desenho de diagrama de objetos.
- d) Implementação
 - Diagrama de instalação;
 - Código fonte;
 - Software funcional.

A fase de implementação envolve a codificação e testes do sistema. Para a aplicação executada no PDA foram utilizadas ferramentas de desenvolvimento para a plataforma Palm OS, dentre emuladores e linguagens próprias. A aplicação executada no PC foi desenvolvida utilizando ferramentas para a plataforma Windows.

A seguir, são descritas as ferramentas utilizadas na construção dos modelos que compõem a análise do sistema (fases anteriores à implementação), bem como no desenvolvimento e implementação do gerenciador remoto.

- 3.1.1. Rational Rose 2000 Enterprise Edition: Trata-se de um conjunto de ferramentas que permite a modelagem visual de diferentes realidades, apresentando detalhes essenciais de um problema complexo e filtrando os detalhes dispensáveis.

O *Rational Rose* foi utilizado no processo de descrição gráfica e elaboração dos diversos diagramas que compõem a modelagem do sistema.

- 3.1.2. Visual Basic 6.0 Enterprise Edition: É um ambiente de desenvolvimento integrado (IDE – *Integrated Development Environment*), no qual é possível desenvolver, executar, testar e depurar aplicativos para a plataforma Windows. A linguagem nativa desse ambiente de desenvolvimento é o VBA (*Visual Basic for Applications*).

A versão *Enterprise Edition* do VB inclui recursos como ferramentas para desenvolvimento de controles *ActiveX* e para a Internet, além de ferramentas como o *Visual SourceSafe* (para controle de versão) e o *Automation e Component Manager*.

A aplicação que roda no PC foi totalmente desenvolvida em *Visual Basic 6*, incluindo rotinas de acesso às portas paralela e serial, essenciais para a comunicação com o controlador digital e o PDA, respectivamente.

- 3.1.3. Corel Draw: Aplicativo gráfico que gera desenhos vetoriais. Ele cria imagens usando formas de objetos armazenados internamente como equações matemáticas (vetor), bem diferente das aplicações que utilizam *bitmap*, uma imagem composta por uma série de pontos.

O Corel Draw foi utilizado na elaboração da tela principal da aplicação do PC, uma imagem elaborada com os efeitos e opções de fundo transparente desta ferramenta, dando à aplicação um visual bonito e moderno.

- 3.1.4. CodeWarrior IDE: Para a aplicação do PDA foi usado o CodeWarrior versão 9.0, IDE de desenvolvimento de software para a plataforma Palm OS. Sua linguagem nativa é o C/C++; sendo que, a aplicação foi totalmente implementada em C.

O CodeWarrior contém um administrador de projetos, um editor de código-fonte, um pesquisador de classes, vários compiladores e linkadores e um depurador de erros integrado. O administrador de projetos organiza todos os arquivos e ambiente relacionado a um projeto, fornecendo uma rápida visualização, fácil organização e navegação entre os arquivos fonte do projeto.

O depurador do CodeWarrior permite o controle do programa em execução, visualizando dados internos enquanto o programa é executado. Ótimo para encontrar problemas em tempo de execução ou após a suspensão da execução de um específico ponto de parada.

3.1.5. Constructor for Palm OS: O Constructor para Palm OS é uma ferramenta de desenho gráfico que acompanha o CodeWarrior e permite a construção de elementos visuais para uma aplicação Palm OS, incluindo: formulários, texto de ajuda, alertas, ícones e figuras.

3.1.6. Palm OS Simulator: O Palm OS Simulator é uma ferramenta de auxílio no desenvolvimento de aplicações para Palm. Uma réplica perfeita do Palm OS, onde as aplicações para esta plataforma podem ser testadas com segurança antes da instalação ou colocação em produção.

O Palm OS Simulator, como o nome sugere, simula um equipamento sob a plataforma Palm OS, sendo utilizada nas fases iniciais de teste da aplicação do PDA.

3.1.7. Palm OS SDK for CodeWarrior: O Palm OS SDK é uma coleção de bibliotecas e cabeçalhos para construção de aplicações para a plataforma Palm OS.

As bibliotecas contidas no SDK são uma cópia das APIs (*Application Programming Interface*) do sistema operacional Palm OS. O próprio CodeWarrior já vem com essas bibliotecas, no entanto, baixou-se da Internet a versão atualizada dessas bibliotecas, incluindo o novo *Serial Manager* (SerialMgr.h), contendo funções utilizadas no estabelecimento da comunicação serial entre o PDA e o PC.

3.2. Controlador Digital

Este dispositivo é uma peça de hardware cuja função é controlar qual equipamento eletrônico deve ser acionado. Este controlador recebe informações oriundas do PC via sua porta paralela, aciona os equipamentos desejados e deve mantê-los acionados até receber comando contrário. Trata-se pois de um dispositivo de comunicação com PC via porta paralela com capacidade de entender os comandos e com capacidade de armazená-los por um tempo indefinido.

Para a construção do controlador digital foram aplicados alguns dos conceitos estudados em lógica digital. Dispositivos eletrônicos (*latches*, demultiplexadores e portas lógicas) foram projetados com o estudo de fundamentos da eletrônica aplicada aos circuitos digitais. Todas as informações necessárias foram coletadas em livros, revistas técnicas da área e *datasheets* dos componentes.

Seguiu-se uma abordagem convencional de desenvolvimento de *hardware*. Primeiramente foi desenhado um diagrama lógico que representasse as funcionalidades desejadas. Uma vez validado este diagrama pode-se definir os componentes que seriam utilizados e desenhar as interconexões entre eles. Foi então montado um protótipo em um *protoboard* que serviu para confirmar sua funcionalidade.

Numa segunda fase partiu-se para o desenvolvimento de um circuito impresso. Este foi desenhado usando uma ferramenta de CAD convencional e foi confeccionado em uma placa de cobre. O circuito foi então montado neste circuito impresso e foi testado.

Na prática, foram utilizados para sua construção os seguintes componentes eletrônicos:

- a) 1 circuito integrado 74LS541 (buffer octal);
- b) 4 circuitos integrados 74LS259 (latches endereçáveis de 8 bits);
- c) 8 resistores de 2,2K Ω ;
- d) 8 resistores de 470 Ω ;
- e) 8 transistores NPN BD137;
- f) 8 diodos 1N4148;
- g) 8 LEDs comuns de qualquer cor;
- h) 8 relés da Metaltex de 5V ASR1C-5V.

4. RESULTADOS

A primeira versão do sistema encaixou-se naquilo pretendido desde o início do projeto, portanto, os resultados foram satisfatórios. O detalhamento desses resultados será apresentado mais adiante, de acordo com os módulos do sistema.

4.1. Gerenciador Remoto

O gerenciador remoto é responsável pelo controle e visualização dos estados dos diversos aparelhos eletrônicos acoplados ao sistema. Contudo, sua flexibilidade permitiu que os resultados fossem muito além disso, tornando-o também num módulo de entrada a outros sistemas computacionais e de disparo de quaisquer ações no PC, tanto local como remotamente. Pode-se por exemplo acionar uma apresentação em *PowerPoint* usando um *Palm* como controle remoto da apresentação.

Segundo Miller (2001), os dispositivos de comunicação sem fios que utilizam *Bluetooth* podem chegar a um alcance de até 100 metros sob condições favoráveis, de acordo com a classe a qual pertença. Entretanto, os dispositivos embutido em PDAs geralmente têm um alcance de apenas 10 metros. O problema é que para atingir distâncias maiores é necessário um consumo de energia maior. Muitos fabricantes decidem então por utilizar uma opção de menor alcance de forma a economizar energia e prolongar o tempo de uso antes de recarregar as baterias dos PDAs. Por esta razão, o alcance do sistema fica limitado ao menor alcance, ou seja, aquele obtido pela maioria dos PDAs.

Isso atribuiu ao projeto uma preocupação adicional, para casos em que um comando fosse enviado fora do alcance de recepção desses dispositivos. Para tanto, foram estabelecidos critérios de modo a garantir a execução das tarefas e a integridade das informações mostradas pelo sistema, que são elas: a descrição dos canais e estado (ligado/desligado) dos equipamentos controlados por ele.

Essas informações são semelhantes em ambos os lados do sistema a partir do momento em que é estabelecida uma conexão entre as duas aplicações, ou melhor, quando os dispositivos estão dentro do alcance permitido para que haja a comunicação entre elas.

Para assegurar isso, foram definidas regras adicionais. Foi implementado um protocolo de comunicação do tipo pára-e-espera com temporização, onde toda informação enviada é retransmitida se não for confirmada num determinado tempo limite, e toda informação recebida com erro é descartada, forçando a sua retransmissão.

Os resultados obtidos estão em concordância com os resultados esperados quando da análise de requisitos do sistema. Podemos observar que quando o usuário solicita (a partir do PDA) a ativação/desativação de algum canal são executados no PC os procedimentos necessários, conforme a Tabela I, que descreve as condições iniciais de execução, entrada e saída esperada para esse procedimento.

Tabela I. Solicitação remota de ação no PC.

Condições Iniciais	<ul style="list-style-type: none">• O sistema do PC deve estar configurado para receber dados pela porta serial definida como padrão pelo serviço de porta serial do <i>Bluetooth</i>;• O dispositivo <i>Bluetooth</i> do PDA deve estar conectado ao dispositivo <i>Bluetooth</i> do PC.
Entrada	<ul style="list-style-type: none">• Comandos LC (liga canal) ou DC (desliga canal) enviados do PDA ao PC quando o usuário clica num botão a partir do sistema do PDA.
Saída Esperada	<ul style="list-style-type: none">• O PC confirma ao PDA o recebimento dos comandos e atualiza o estado do canal solicitado, em ambas as aplicações (PC e PDA), enviando imediatamente sinais ao controlador digital.

Na Figura 2 é mostrado o resultado, ou seja, a tela do sistema quando o comando “ligar canal 12” (LC12) é disparado a partir do clique no botão de comando “Canal 12” no sistema do PDA, ocasionando a imediata ativação do lado do PC.

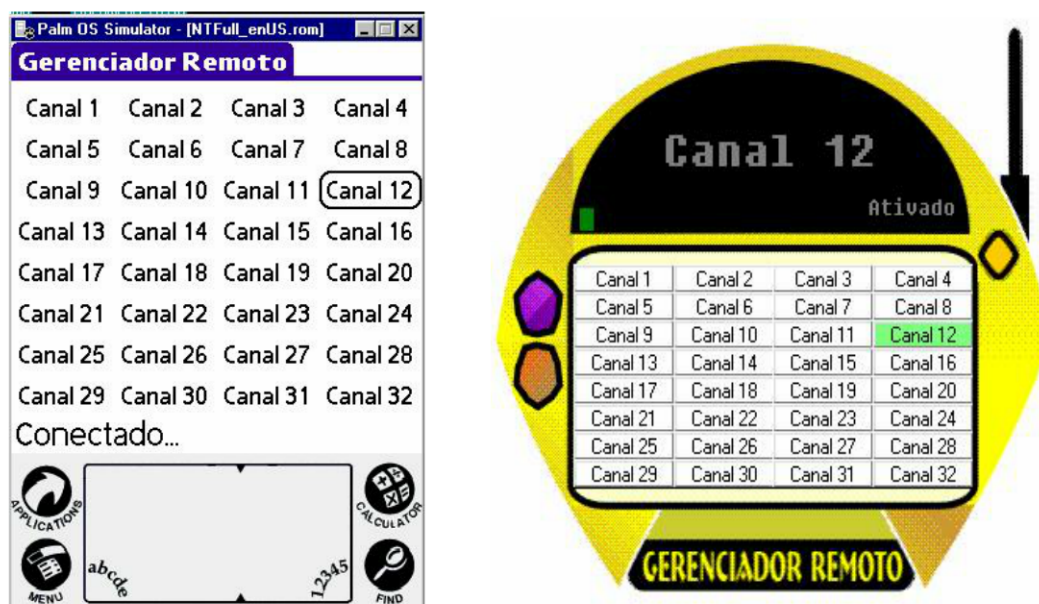


Figura 2. Resultados obtidos com a ativação de um canal.

4.2. Controlador Digital

O controlador digital é o *hardware* responsável pela interface entre o PC e os aparelhos os quais queremos controlar, uma vez que estes aparelhos não possuem meios para estabelecer uma comunicação direta com o PC e muito menos com o PDA, já que não dispõem de dispositivos de comunicação *wireless* e nem seguem a um padrão de comunicação comum entre si.

O controlador digital possui x entradas e n saídas, sendo que n é maior que x , onde x são os 8 bits recebidos do barramento de dados da porta paralela do PC. Após receber esses sinais, o controlador digital executa operações pré-definidas por seus circuitos lógicos, colocando nas suas saídas resultantes uma tensão senoidal alternada capaz de ligar um dos n aparelhos ligados a ele (Figura 3).

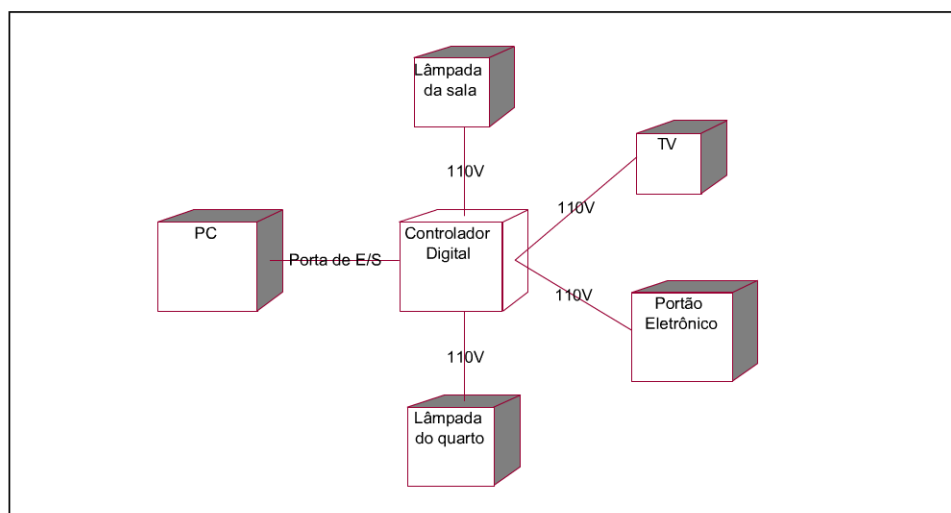


Figura 3. Visão geral das ligações ao controlador digital.

Tendo como base os conceitos fundamentais de lógica digital, ver por exemplo Taub (1982) ou Bogart Jr. (1991), conclui-se que cada sinal recebido pelo controlador digital estabelece um comportamento único às saídas do circuito, sendo enviado uma única vez e assim permanecendo até que outro sinal diferente seja recebido. Um buffer ou *latch* de n bits foi adicionado à entrada deste controlador para armazenar este sinal, e que, além dessa funcionalidade, protege a porta de entrada/saída do computador, evitando sua queima acidental.

O bloco seguinte é o responsável pela ampliação da quantidade de saídas possíveis. É neste módulo que os sinais são demultiplexados, ativando ou desativando, dependendo de um bit de controle, alguma das suas saídas. Bignell e Donovan (1995) mostram que os demultiplexadores não têm a propriedade de armazenar os estados anteriores de suas saídas, presença ou ausência de tensão. Por esta razão pequenos módulos de memória, *latches*, fez-se necessário a fim de manter esses estados, alimentando as chaves que ativam os aparelhos. Na prática, um único componente eletrônico executa essas duas funções (demultiplexador e memória). No circuito, trata-se de um *latch* endereçável de 8 bits.

Por fim, temos várias chaves eletrônicas que ativam os relés que alimentam os aparelhos eletrônicos ligados a eles. A Figura 4 mostra a arquitetura completa do controlador digital, separando-o em blocos.



Figura 4. Arquitetura do controlador digital.

O barramento de dados da porta paralela do computador possui 8 bits (D0, D1, D2, D3, D4, D5, D6 e D7). Cada um desses bits desempenha uma função específica no controlador digital.

Conforme a Tabela II, os bits D0, D1 e D2 são usados para selecionar uma das oito saídas de cada *latch* endereçável de 8 bits da Tabela II. Como é mostrado na Tabela IV, o valor resultante (Y) de cada uma dessas combinações é somado a outros valores e só então são definidos os canais ativos do circuito.

Tabela II. Combinação binária de seleção de saídas dos *latches* endereçáveis.

Seleciona Saída			
D2	D1	D0	Y (Saída)
0	0	0	1
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

O bit D3 é aplicado às saídas selecionadas segundo a Tabela acima, onde 1 indica a presença de tensão e 0 a ausência de tensão, ou seja, é esse bit que indica se o aparelho conectado ao canal será ligado ou desligado, respectivamente. A Tabela III ilustra o que foi dito.

Tabela III. Valor binário para ativação do canal.

Ativa/desativa canal	
D3	Ação
1	Ativa canal
0	Desativa canal

A figura 8 mostra a combinação de bits aplicada às entradas D4, D5, D6 e D7 usada para selecionar um dos quatro *latches* endereçáveis do circuito de forma individual ou simultânea. Numa análise geral, o nível lógico 0 aplicado a D4, D5, D6 ou D7 seleciona um dos quatro CIs associados a cada uma dessas entradas, o inverso também ocorre, o que significa que 0, 1, 2, 3 ou 4 canais podem ser acionados simultaneamente.

Tabela IV. Combinação binária de seleção dos *latches* endereçáveis.

Seleciona CI				CI Selecionado			
D4	D5	D6	D7	CI1	CI2	CI3	CI4
0	0	0	0	Y	Y+8	Y+16	Y+24
0	0	0	1	Y	Y+8	Y+16	
0	0	1	0	Y	Y+8		Y+24
0	0	1	1	Y	Y+8		
0	1	0	0	Y		Y+16	Y+24
0	1	0	1	Y		Y+16	
0	1	1	0	Y			Y+24
0	1	1	1	Y			
1	0	0	0		Y+8	Y+16	Y+24
1	0	0	1		Y+8	Y+16	
1	0	1	0		Y+8		Y+24
1	0	1	1		Y+8		
1	1	0	0			Y+16	Y+24
1	1	0	1			Y+16	
1	1	1	0				Y+24
1	1	1	1				

☒ CI selecionado
☐ CI não selecionado

Cada CI selecionado possui um valor associado a ele que, somado ao valor resultante da figura 6, indica quais serão os canais selecionados pelo controlador digital, ou seja, os canais selecionados pelo circuito dependem da combinação dos resultados obtidos nas figuras 6 e 8. Com isso, segundo cada um dos valores aplicados à saída da porta paralela do PC, é possível verificar quais são os resultados obtidos. Por exemplo, lembrando que os bits mais significativos estão à esquerda de qualquer combinação binária, o valor binário 11101000 apresenta o seguinte resultado:

D0=0, D1=0 e D2=0, logo, Y=1.

D3=1, logo, o canal é ligado.

D4=0, D5=1, D6=1 e D7=1, logo, apenas o CI1 é ativado. Como CI1=Y, ou melhor, CI1=1, apenas o canal 1 do controlador digital é selecionado e o nível lógico 1 é aplicado à saída desse canal, o que na prática ligaria o aparelho acoplado a esse canal.

Por fim, temos o custo total do projeto, incluindo todos os itens necessários para sua realização (Tabela V)

Tabela V. Orçamento do projeto.

EQUIPAMENTOS ADQUIRIDOS	
Adaptador USB <i>Bluetooth</i> , modelo BTM200B	R\$134,00
COMPONENTES P/ CONSTRUÇÃO DO CONTROLADOR DIGITAL	
Buffer e latches endereçáveis	
Componentes eletrônicos (CIs)	R\$7,00
Chave eletrônica (protótipo com 8 entradas)	
Componentes eletrônicos (resistores, transistores, diodos e relés)	R\$53,00
Outros itens	
Placa de circuito impresso	R\$8,00
Solda	R\$1,00
Fios, cabos e conectores	R\$7,00
Fonte de alimentação de 5 volts	R\$18,00
Total	R\$228,00

5. CONCLUSÃO

A comunicação sem fio representa uma grande vantagem, uma vez que cabos, fios e conectores são dispensados, porquanto eles poderiam constituir-se em séria complicação em virtude da grande variedade existente no mercado. Contudo, em equipamentos eletrônicos de uso doméstico, a comunicação por infravermelho ainda é a mais comum, porém, seu curto alcance e impossibilidade de atravessar obstáculos sólidos, além de sua característica de visada direta inviabilizam o seu uso em aplicações mais sofisticadas.

A tecnologia de comunicação *Bluetooth*, em comparação com outras tecnologias sem fio, foi a que se apresentou mais atraente no que diz respeito ao consumo de energia, ao alcance obtido e à quantidade de equipamentos portáteis integrados com ele - presente não apenas em *palmtops* como em vários equipamentos eletrônicos, inclusive em telefones celulares. Além disso, por se tratar de um padrão mundial, o *Bluetooth* garante que a tecnologia utilizada em qualquer lugar do mundo é a mesma, bastando que os dispositivos estejam próximos para se comunicarem.

O aqui projeto apresentado é composto por um gerenciador remoto e por um controlador digital. Todo sistema foi desenvolvido no CEFET. A parte lógica, ou melhor, o programa de computador usado para controlar o estado dos equipamentos ligados à parte física é facilmente configurável, tornando-o um módulo independente deste último. Na verdade, o gerenciador remoto pode interagir com qualquer equipamento que se conecte às portas paralela e/ou serial do computador, além de poder ser usado na execução de ações diretamente no computador e interagir com outros programas. Essa característica estende o seu uso, não o limitando apenas a um tipo de aplicação.

O PDA possui a tarefa de controlar as ações à distância. A sua aquisição representou um custo relativamente alto. No entanto, projetos futuros indicam os telefones celulares como solução para minimizar ainda mais os custos e também abranger um maior número de pessoas interessadas nesta solução, uma vez que a disseminação deles no mercado é bem maior.

Por fim, embora o tempo tenha sido um fator determinante, o produto final não deixa a desejar em qualidade e oferece mais uma opção de automação residencial para os amantes de tecnologia e para aqueles que necessitam desse tipo de solução.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- Alves, William Pereira. **Palm OS e Windows CE: desenvolvimento de aplicações**. São Paulo: Érica, 2002.
- Bignell, James W. e Donovan, Robert L. **Eletrônica digital: lógica sequencial**. v.2. São Paulo: Makron Books, 1995.
- Bogart Jr., Theodore F. **Introduction to Digital Circuits**. Macmillan / McGraw-Hill, 1992.
- Bona, Cristina e Costa, Marcello Thiry C. **Processo de software: um estudo de caso em Iconix**. Disponível em: <www.cbcomp.univali.br/anais/pdf/2003/eng123.pdf> Acesso em: 17 maio 2004.
- Kruchten, Philippe. **The Rational Unified Process: An Introduction**. Reading: Addison Wesley Longman, 1999.
- Leffingwell, Dean e Widrig, Don. **Managing Software Requirements: A Unified Approach**. Reading: Addison-Wesley, 2000.
- Miller, Michael. **Descobrendo Bluetooth**. Rio de Janeiro: Campus, 2001.
- Rumbaugh, James; Jacobson, Ivar e Booch, Grady. **The Unified Modeling Language Reference Manual**. Reading: Addison Wesley Longman, 1998.
- Sintes, Anthony. **Aprenda Programação Orientada a Objetos em 21 Dias**. São Paulo: Pearson Education do Brasil, 2002.
- Taub, Herbert e Schilling, Donald. **Eletrônica Digital**. Editora McGraw Hill do Brasil, 1982.
- Ambler, Scott e Jeffries, Ron. **Agile Modeling: Effective Practices for Extreme Programming and the Unified Process**. Wiley, 2002.