



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

PTC2531 – Laboratório de Projeto de Formatura II

SISTEMA PARA VISUALIZAÇÃO REMOTA DE ELETROCARDIOGRAMA VIA CELULAR

Relatório Final de Projeto de Formatura

PTC - Departamento de Telecomunicações e Controle

LAC – Laboratório de Automação e Controle

Edson Carvalho Mota Júnior
Jairo Paulino Ramalho

Orientador: Prof. Dr. José Carlos Teixeira de Barros Moraes

São Paulo, 05 de dezembro de 2005

PTC
TF-2005
M856s

Índice

Resumo	2
1. Introdução e Justificativas	3
2. Objetivos	4
3. Tópicos Básicos	5
3.1. O Coração e o Sistema Circulatório	5
3.2. O Eletrocardiograma	8
4. Descrição Geral e Especificação do Projeto	11
5. Especificação das tecnologias e equipamentos utilizados	13
5.1. Simulador de Sinais Cardíacos	13
5.2. Placa Lynx	13
5.3. A linguagem PHP	13
5.4. O Protocolo WAP e a linguagem WML	15
5.4.1. Wireless Application Protocol (WAP)	15
5.4.2. Wireless Markup Language (WML)	16
6. Programas utilizados	17
6.1. Desenvolvedor em PHP	17
6.2. Apache	17
6.2.1. Configuração do Apache	17
6.3. Desenvolvedor em WML	18
7. Descrição Funcional	20
7.1. Projeto do Software do PC	20
7.1.1. Módulo da placa	20
7.1.2. Módulo de captura	20
7.1.3. Módulo de geração da imagem	20
7.2. Projeto do Software do Servidor	22
7.2.1. Especificações do software	22
7.2.2. A página WAP	22
7.2.2.1. Cenário I	22
7.2.2.1.1. Deck 1 - Apresentação	22
7.2.2.1.2. Deck 2 - Figuras de ECG	24
7.2.2.2. Cenário II	24
7.2.2.2.1. Deck 1 - Apresentação:	24
7.2.2.2.2. Deck 2 - Figuras de ECG	26
8. Testes e Validação	27
8.1. Aquisição dos dados	27
8.2. Geração da figura	32
8.3. Transferência e visualização da figura no celular e na página Web	37
9. Considerações finais	42
Agradecimentos	43
Referências Bibliográficas	44
Anexo A - Cronograma	45
Anexo B - Códigos-fonte Software PC	46
Anexo C - Códigos-fonte Software Servidor	63
Anexo D - Códigos para controle do ECGSim via HyperTerminal	69

Resumo

O presente relatório tem o objetivo de apresentar uma descrição detalhada do Projeto de Formatura desenvolvido pelo grupo, ***Sistema de Visualização Remota de Eletrocardiograma via celular***. Nas próximas páginas são abordados sua concepção, os equipamentos e programas utilizados, a definição das tecnologias, a descrição funcional e a integração e os testes de validação do resultado obtido com o projeto.

Conjuntamente, serão apresentados os códigos desenvolvidos e as características principais das tecnologias e dos programas de apoio ao desenvolvimento utilizados.

1. Introdução e Justificativas

Para determinados pacientes com doenças crônicas, como problemas cardíacos, faz-se necessária uma constante monitoração de seu eletrocardiograma (ECG). A ida constante ao consultório médico pode se tornar maçante já que o paciente tem que se locomover constantemente ao consultório. Além do inconveniente da locomoção do paciente, o médico tem de dispor um horário em sua agenda para atender pessoalmente o paciente a fim de avaliar o seu ECG. Para minimizar esses efeitos, torna-se útil um sistema de visualização remota de eletrocardiograma, com o qual um médico à distância pode visualizar o ECG do paciente. O paciente não necessita sair do conforto de sua residência e o médico pode analisar o ECG no momento em que achar mais conveniente, entre consultas ou em resposta a um pedido do paciente, mesmo que esteja longe de seu consultório, por meio de seu celular.

2. Objetivos

O objetivo das disciplinas “Projeto de Formatura I e II” é incumbir os alunos do desenvolvimento de um projeto, com o intuito de consolidar os conhecimentos adquiridos na área de Engenharia Elétrica e de prepará-los para a vida profissional futura.

Durante o desenvolvimento do projeto, os alunos estiveram envolvidos com a definição do escopo do projeto, pesquisa de diferentes tecnologias, planejamento, implementação, testes e documentação das soluções.

Os seguintes ramos das áreas de Engenharia Elétrica foram abordados durante o desenvolvimento do projeto:

- Eletrônica analógica e digital
- Interface de comunicação RS-232
- Projeto e desenvolvimento de softwares em linguagem PHP
- Geração de figuras através de um arquivo de dados
- Fisiologia humana e Engenharia Biomédica
- Comunicação pelo Protocolo WAP
- Desenvolvimento de páginas em WML.
- Integração de sistemas
- Especificação, planejamento e desenvolvimento de um projeto

3. Tópicos Básicos

Para entendermos melhor de que forma e com qual resolução o médico necessita do sinal eletrocardiográfico para análise, realizamos um estudo referente ao sistema circulatório e o eletrocardiograma.

3.1. O Coração e o Sistema Circulatório

O sangue tem a tarefa de transportar para todas as células de nosso organismo oxigênio e materiais nutritivos. O sangue também recebe das células as substâncias de rejeição, como o gás carbônico, o qual será levado ao pulmão para trocas gasosas serem realizadas. Para realizar essas tarefas, o sangue tem necessidade de "circular" continuamente: os canais dentro dos quais o sangue circula são os vasos sangüíneos, enquanto o coração é a bomba que dá ao sangue o seu impulso para circulação. Coração e vasos constituem, no seu conjunto, o aparelho circulatório.

Para bombear o sangue através das artérias para todo o corpo e para manter o controle dos vasos que chegam ao coração, ocorre o que denominamos ciclo cardíaco, onde cada fase deve acontecer da forma certa, em uma seqüência precisa e no instante correto.

O sistema responsável pela geração e condução do impulso é formado por um conjunto de células, estrategicamente distribuído pelo coração, cujo limiar de excitação e capacidade de condução do impulso é diferente do tecido muscular vizinho. Abaixo segue uma figura do coração humano representando suas principais partes, as quais estão intimamente envolvidas com o ciclo cardíaco.

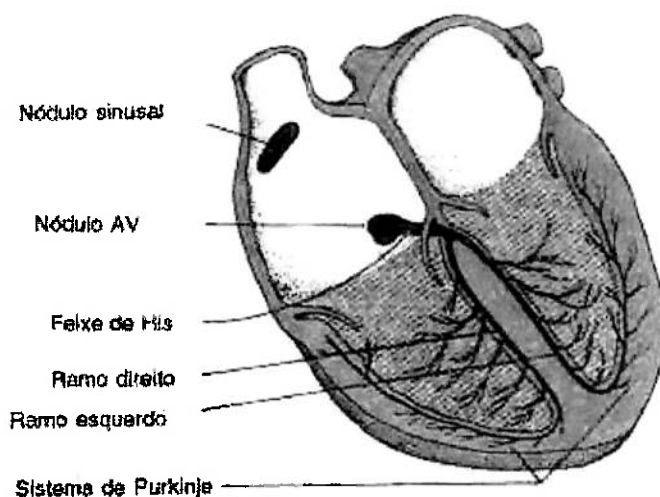


Fig. 2.3 Sistema de condução.

Figura 1 – O coração humano

Para possibilitarmos o ciclo cardíaco, três eventos interdependentes, mas não simultâneos devem ocorrer: a despolarização celular; a contração miocárdica, que

constituirá o ciclo fisiológico; e o movimento cinético do sangue e das valvas cardíacas que compõe o ciclo cardíaco propriamente dito. Apenas este último é o que determina o que estudamos como ciclo cardíaco. Assim devemos compreender que os eventos contráteis, são precedidos dos eventos de excitação elétrica e, por sua vez, os eventos hemodinâmicos sucedem os eventos contráteis. Só assim entenderemos os conceitos de sístole e diástole descritos abaixo. Para fins didáticos iniciamos o estudo no momento final da diástole ventricular, a sístole atrial.

A ativação elétrica do coração

A contração do músculo cardíaco depende diretamente da despolarização elétrica das células miocárdicas. O estímulo elétrico se origina nas células marca-passo do nó sino-atrial. A despolarização elétrica caminha pelo músculo atrial e chega ao nó atrioventricular, onde sofre um retardo fisiológico. Este retardo é que vai originar a contração seqüencial do átrio e ventrículo de forma harmônica. Teoricamente os átrios e ventrículos são isolados eletricamente, possuindo como ligação normal apenas o nó AV. Falhas nesse isolamento podem ocorrer através da existência de bandas musculares que atravessam o anel fibroso AV, formando as chamadas vias anômalas, responsáveis pela Síndrome de Wolf Parkinson White e por certas taquiarritmias. O estímulo elétrico se propaga aos ventrículos através do feixe de His e seus ramos, distribuindo-se por todo o miocárdio ventricular através das fibras de purkinje. A despolarização do sistema de condução obviamente antecede a contração mecânica.

A contração isovolumétrica

Após a sístole atrial os ventrículos estão em sua capacidade máxima de volume e com sua maior pressão diastólica (pressão diastólica final ou PD2). As valvas aórtica e pulmonar encontram-se fechadas, pois as pressões diastólicas arteriais são bem maiores ainda que a pressão diastólica dos ventrículos. A ativação elétrica chega ao ventrículo e começa a fase de contração muscular. A pressão intracavitária sobe rapidamente e as valvas AV se fecham completamente. Como as pressões atriais se assemelham muito e o ventrículo esquerdo, por ser maior, produz mais rapidamente maior pressão, a valva mitral se fecha primeiro que a tricúspide. Por um momento extremamente curto as cavidades ventriculares se isolam completamente, enquanto a contração do músculo ventricular se propaga. Em um certo ponto a pressão intracavitária ultrapassa a pressão diastólica das grandes artérias e as valvas semilunares se abrem.

A fase de ejeção

Com a abertura das valvas semilunares ocorre a expulsão do volume ventricular para as grandes artérias. A contração do músculo cardíaco chega ao seu máximo no ápice da curva de pressão ventricular, que depois começa a se reduzir. A taxa da ejeção ventricular não depende apenas da força de contração do músculo cardíaco e do gradiente de pressão formado a nível das valvas, mas também das propriedades elásticas dos grandes vasos e da árvore arterial.

O relaxamento e as fases de enchimento ventricular

Se pensarmos com atenção, devemos entender que com o fim da contração muscular e o início da diástole fisiológica, na verdade ainda estamos no período da sístole

cardíaca. A contração começa a se extinguir, com cada vez mais miofibrilas em estado de relaxamento, a partir do ápice da onda de pressão ventricular. A pressão se reduz demorando ainda um certo período de tempo onde a ejeção ainda ocorre mas numa fase reduzida, mesmo com o músculo cardíaco já tendo iniciado seu relaxamento. Apesar do fluxo sanguíneo dos ventrículos para os grandes vasos diminuir muito, o fluxo sanguíneo na árvore arterial se mantém devido as propriedades elásticas das paredes dos grandes vasos (efeito Windkessel). Em um determinado ponto a pressão arterial supera a ventricular, fechando as valvas aórtica e pulmonar. A diástole cardíaca começa quando estas valvas se fecham. Nesse instante ocorre o entalhe dicrótico da curva de pressão arterial.

O relaxamento isovolumétrico

O músculo cardíaco já com a fase de relaxamento iniciada gera uma queda repentina da pressão intracavitária. Em um dado momento a pressão arterial supera a pressão ventricular e as valvas se fecham, primeiro a aórtica por possuir maior pressão e em seguida a pulmonar. Durante alguns instantes a pressão ventricular permanece maior que a atrial e ambas as vias de entrada e saída permanecem fechadas, apesar do relaxamento ativo das fibras miocárdicas.

A diástole

a) A fase de enchimento rápido

Quando a pressão ventricular por fim se reduz abaixo da pressão atrial, que nesse momento é máxima (ápice da onda v da curva de pressão atrial - linha 4 - fig1) as valvas AV se abrem deixando passar um grande fluxo rapidamente em direção ao ventrículo. 70% do enchimento ventricular ocorre nessa fase.

b) Fase de enchimento lento

Também chamado de diastase. Com o enchimento do ventrículo e o fim da fase ativa do relaxamento do músculo cardíaco, ocorre uma desaceleração importante do fluxo. As valvas AV tendem a se fechar passivamente. No momento da desaceleração do fluxo rápido para o fluxo lento é que ocorre o 3º ruído cardíaco. O fluxo do átrio para o ventrículo é bastante reduzido, chegando a quase parar.

c) Sístole atrial

Nova contração atrial ocorre, finalizando o ciclo. As valvas AV se reabrem, momento em que ocorre a onda A da valva mitral ao ECO unidimensional e o 4º ruído cardíaco. A sístole atrial pode representar até 20% do volume diastólico final do ventrículo, sendo de grande importância para a manutenção do débito cardíaco nos pacientes que possuam algum tipo de restrição funcional do VE.

3.2. O Eletrocardiograma

O Eletrocardiograma pode ser definido como um gráfico de variações de voltagem em relação ao tempo. As variações resultam da despolarização e repolarização do músculo cardíaco, que por sua vez produz campos elétricos, que atingem a superfície do corpo onde os eletrodos estão localizados.

Durante a Polarização Cardíaca, o músculo cardíaco em repouso conserva íons + do lado externo e íons - do lado interno da membrana celular.

Durante a Despolarização Cardíaca, ocorre a inversão da distribuição de íons espontânea ou por estímulo elétrico externo.

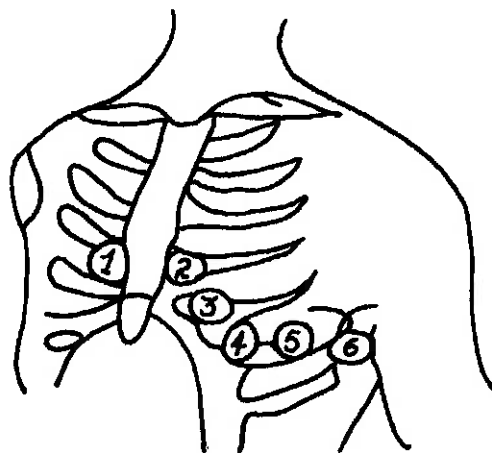
Na superfície do corpo existem diferenças de potencial, conseqüentes aos fenômenos elétricos gerados durante a excitação cardíaca. Estas diferenças podem ser medidas e registradas tendo-se uma noção satisfatória do tipo e da intensidade das forças elétricas do coração. Para isto são utilizados galvanômetros de tipo particular que constituem as unidades fundamentais dos eletrocardiógrafos. Nesse sentido os pontos do corpo a serem explorados são ligados ao aparelho de registro por meio de fios condutores (eletrodos). Dessa forma, obtêm-se as chamadas derivações que podem ser definidas de acordo com a posição dos eletrodos. Ela é dita bipolar quando os dois eletrodos se encontram a mesma distância - do ponto de vista elétrico - do coração. Derivações unipolares são aquelas em que o traçado obtido se deve às variações de potencial recolhidos por um dos eletrodos dito explorador. O outro fica relativamente muito mais distante do coração, situado num ponto de potencial zero denominado eletrodo indiferente. Eletrodo explorador e indiferente são também utilizados para as derivações bipolares, sendo o primeiro o eletrodo positivo e o segundo o eletrodo negativo. Por convenção registram-se curvas ditas positivas quando um dos eletrodos admitido como explorador está orientado para as áreas que se comportam como positivas em relação às que se encontram voltadas para o outro eletrodo (eletrodo indiferente).

As Derivações Eletrocardiográficas

Plano Frontal: Standart D1, D2, D3
Unipolares aVR, aVL, aVF

Plano Horizontal: Precordiais
V1, V2, V3, V4, V5, V6

Derivações no Plano Frontal:
D1 - diferença de potencial entre braço esquerdo e direito
D2 - diferença de potencial entre pernas e braço direitos
D3 - diferença de potencial entre pernas e braço esquerdos



DERIVAÇÕES PRECORDIAIS DE V₁ A V₆

Derivações no Plano Horizontal:

V1 - 4º espaço intercostal, bordo direito esterno
 V2 - 4º EI, bordo esquerdo esterno
 V3 - entre V2 e V4
 V4 - 5º EI, linha hemiclavicular
 V5 - 5º EI, linha axilar anterior
 V6 - 5º EI, linha axilar média

Na eletrocardiografia prática, se aceita como verdadeiras as seguintes proposições:

- O tronco humano é uma esfera condutora homogênea, de tamanho infinito.
- O coração está no centro, ocupando o tamanho de um ponto.

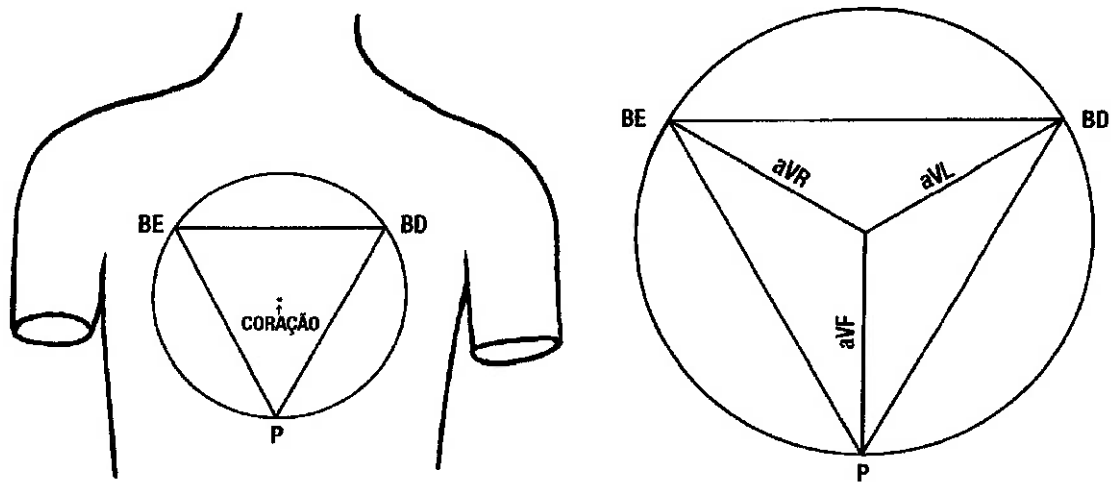


Figura 2 – O triângulo de Einthoven

Assim, o conceito de Triângulo de Einthoven pressupõe que a união das extremidades (brazos e perna) forma um triângulo equilátero, e que o coração considerado um ponto, ocupe o centro do triângulo e da esfera (tronco).

Desta forma, podemos aplicar algumas leis físicas de fluxo elétrico:

Então temos:

- aVR, aVL e aVF são potenciais absolutos do BD, BE e perna em relação ao coração, que está no centro do triângulo.

Com base nas derivações obtidas, podemos gerar a figura a seguir, que apresenta o formato de uma ciclo de um sinal de eletrocardiograma, que nada mais é do que a resultante dos potenciais das diversas partes do coração, em função do tempo para um ciclo cardíaco normal.

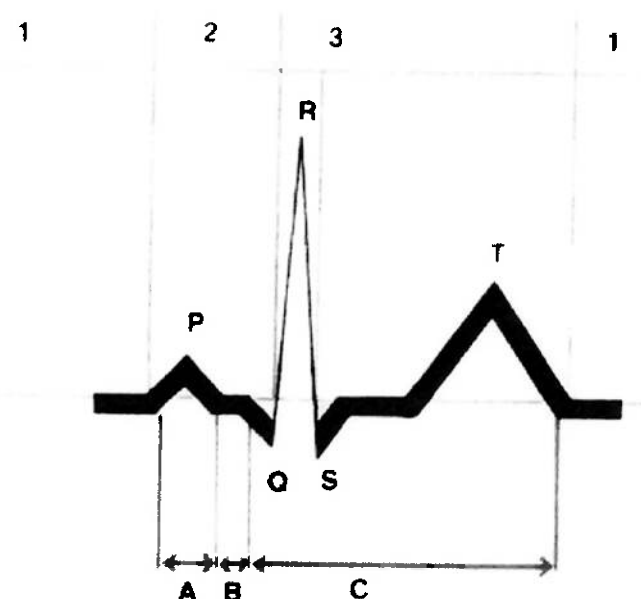


Figura 3 – Resultante da atividade elétrica cardíaca

Potencial Elétrico – diferença de concentração iônica.

Onda P – Despolarização Atrial

Intervalo P-R – Intervalo de Tempo, começo da despolarização atrial até começo da despolarização ventricular

Complexo Ventricular QRS – Despolarização dos Ventrículos

Onda Q – Despolarização Septal (Deflexão P/A Baixo)

Onda R – Despolarização Ventricular (Deflexão P/A Cima)

Onda S – 1ª Deflexão Negativa seguinte a onda R. Despolarização da região basal posterior do ventrículo E.

Onda T – Repolarização dos Ventrículos.

Segmento S-T – Período de inatividade elétrica depois do miocardio estar despolarizado.

Onda U – Segue a onda T originada pelos potenciais tardios do início da diástole.

Intervalo Q-T – Tempo necessário para despolarização e repolarização dos ventrículos.

4. Descrição Geral e Especificação do Projeto

A Figura 1 apresenta uma descrição gráfica do sistema desenvolvido. O sistema tem como entrada um sinal de eletrocardiograma fornecido por um simulador de ECG. O sinal gerado é enviado para um computador localizado na residência do paciente através da interface serial. Por meio de um software deste PC, uma figura é gerada com base no sinal oriundo do simulador. Essa figura é enviada a um servidor Web. No servidor uma interface foi desenvolvida para permitir que o médico possa acessar as informações do ECG do paciente em um PC conectado à internet ou via WAP por meio de um celular, com os quais pode fazer sua análise da forma de onda capturada.

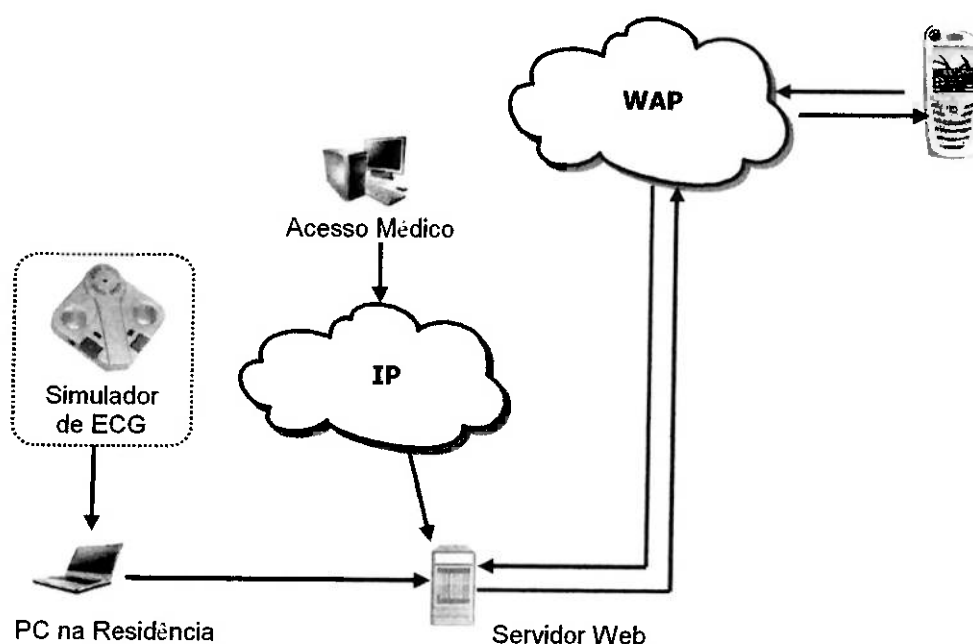


Figura 4 – Sistema de Visualização Remota de ECG via celular

Glossário:

ECG: Eletrocardiograma

WAP: Wireless Application Protocol

IP: Internet Protocol

O Simulador de ECG foi obtido junto ao Laboratório de Engenharia Biomédica. Ele foi elaborado durante a tese de mestrado de José Gomes Gonçalves Filho intitulada “Simulador de sinais cardíacos” [2], apresentada à Escola Politécnica da USP em 1999.

O PC na Residência e o Servidor Web são baseados em Windows e podem ser fisicamente alocados em uma mesma máquina, desde que virtualmente formem plataformas distintas.

O médico pode acessar remotamente as informações armazenadas no servidor Web por meio da internet, digitando o endereço IP do servidor e utilizando um browser adequado.

A rede WAP é de propriedade das operadoras de telefonia móvel e permite o acesso de dados em velocidades em torno de 70kbps.

O equipamento celular do médico deve ser habilitado a acessar a rede WAP da operadora. Deve apresentar um browser adequado e ser capaz de visualizar imagens nos formatos gif e wbmp. Apenas uma pequena minoria dos celulares presentes no mercado atualmente não possui essas características. Assumindo uma constante evolução dos aparelhos móveis para os próximos anos, garantimos que quase a totalidade dos médicos que pretendem utilizar o sistema possuam celulares que tenham os requisitos tecnológicos mínimos para tanto.

5. Especificação das tecnologias e equipamentos utilizados

5.1. Simulador de Sinais Cardíacos

O Simulador de Sinais cardíacos, denominado ECGSim, é composto de uma pequena caixa contendo os circuitos elétricos em seu interior e um programa de controle executado em um computador PC ligado ao simulador por uma interface RS-232 padrão.

A caixa do ECGSim permite a ligação em monitores cardíacos e equipamentos de eletrocardiograma com capacidade para até 10 terminais usuais: RA, LA, RL, LL além de V1-V6, referencia na perna direita RL. Na parte posterior da caixa há um conector para a interface serial de comunicação com o microcomputador do PC. A alimentação é feita por uma bateria de 9V interna com conector para fonte de alimentação externa. Em uma das laterais existe uma chave liga-desliga.

O Hardware é controlado por um programa de controle que permite a escolha do sinal a ser gerado e a alteração dos parâmetros pertinentes ao sinal em questão como, por exemplo, frequência cardíaca, escala do gráfico, ajuste de amplitude, etc. O programa também permite escolher sinais de arritmias para serem gerados pelo simulador.

Durante a utilização do simulador no projeto de formatura desenvolvido por nosso grupo não utilizamos o programa desenvolvido para o ECGSim, mas sim, utilizamos o Hyper Terminal para enviar os comandos para o simulador via interface serial. Dependendo do comando enviado, definíamos as características da forma de onda desejada. Anexa a esse relatório pode ser encontrada a lista de comandos utilizados para controlar o simulador:

5.2. Placa Lynx

Utilizamos a Placa Lynx [5] e um programa em C auxiliar, cujo código-fonte está anexo a este relatório, para realizarmos a aquisição dos dados e armazenar os sinais captados na forma de arquivos que podem ser consultados posteriormente.

A placa de captura que utilizamos para a aquisição de dados foi a placa Lynx 12/32, que possui como características de conversor A/D: 8, 16 e 32 canais podem ser usados, com uma resolução numérica de 12 bits, podendo chegar a até 20.000 amostras por segundo, e a tensão das entradas e das saídas podem chegar até ± 5 V.

Entre outros recursos que esta placa possui, podemos destacar: possui 16 entradas digitais TTL e 16 saídas digitais TTL, além de ter suporte a IRQ (são interrupções de hardware, canais que os dispositivos podem utilizar para chamar a atenção do processador) e possuir 3 contadores para base de tempo e uso geral.

5.3. A linguagem PHP

O PHP é uma linguagem para programar scripts do lado do servidor, que se incrustam dentro do código HTML. Esta linguagem é gratuita e de multiplataforma.

PHP é o acrônimo de Hipertext Preprocesor. É uma linguagem de programação do lado do servidor gratuito e independente de plataforma, rápido, com uma grande livreria de funções e muita documentação.

Uma linguagem do lado do servidor é aquela que se executa no servidor web, justo antes da página ser enviada através da Internet ao cliente. As páginas que se executam no servidor podem realizar acessos a bases de dados, conexões em rede, e outras tarefas para criar a página final que será vista pelo cliente. O cliente somente recebe uma página com o código HTML resultante da execução da PHP. Como a página resultante contém unicamente código HTML, é compatível com todos os navegadores.

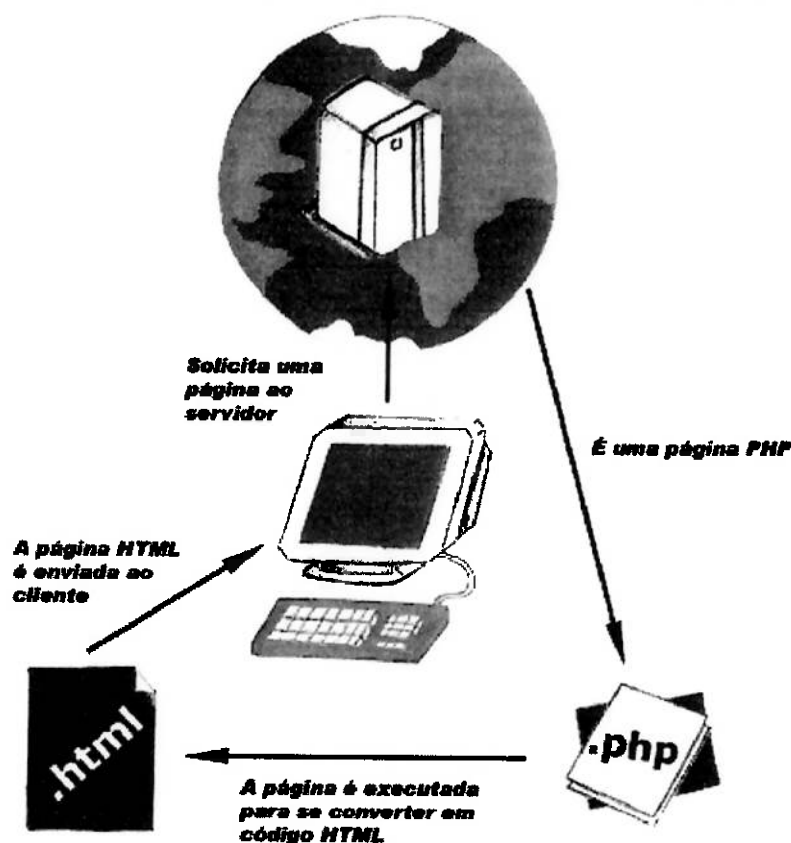


Figura 5 - Esquema do funcionamento das páginas PHP.

Uma vez que já conhecemos o conceito de linguagem de programação de scripts do lado do servidor podemos falar de PHP. PHP se escreve dentro do código HTML, o que o faz realmente fácil de utilizar, assim como ocorre com o popular ASP de Microsoft, porém com algumas vantagens como sua gratuidade, independência de plataforma, rapidez e segurança. Qualquer pessoa pode baixar através da página principal de PHP www.php.net e de forma gratuita, um módulo que faz com que nosso servidor web compreenda os scripts realizados nesta linguagem. É independente de plataforma, visto que existe um módulo de PHP para quase qualquer servidor web. Isto faz com que qualquer sistema possa ser compatível com a linguagem e significa uma vantagem importante, já que permite levar o site desenvolvido em PHP de um sistema a outro sem praticamente nenhum trabalho.

PHP, no caso de estar montado sobre um servidor Linux ou Unix, é mais rápido que ASP, dado que se executa em um único espaço de memória e isto evita as comunicações entre componentes COM que se realizam entre todas as tecnologias implicadas em uma página ASP.

Por último assinalamos a segurança, neste ponto também é importante o fato de que em muitas ocasiões PHP se encontra instalado sobre servidores Unix ou Linux, que são bastante conhecidos como mais velozes e seguros que o sistema operativo onde se executa as ASP, Windows NT ou 2000. Ademais, PHP permite configurar o servidor de modo que se permita ou rejeita diferentes usos, o que pode fazer da linguagem mais ou menos segura dependendo das necessidades de cada um.

Foi criado originalmente em 1994 por Rasmus Lerdorf, mas como PHP está desenvolvido em política de código aberto, ao longo de sua história teve muitas contribuições de outros desenvolvedores. Atualmente PHP se encontra em sua versão 4, que utiliza o motor Zend, desenvolvido com maior meditação para cobrir as necessidades das aplicações web atuais.

Esta linguagem de programação está preparada para realizar muitos tipos de aplicações web graças à extensa livreria de funções com a qual está dotada. A livreria de funções cobre desde cálculos matemáticos complexos até tratamento de conexões de rede, podemos dar dois exemplos.

Algumas das mais importantes capacidades de PHP são: compatibilidade com as bases de dados mais comuns, como MySQL, mSQL, Oracle, Informix, e ODBC, por exemplo. Inclui funções para o envio de correio eletrônico, upload de arquivos, criar dinamicamente no servidor imagens no formato GIF, inclusive animadas e uma lista interminável de utilidades adicionais.

5.4. O Protocolo WAP e a linguagem WML

5.4.1. Wireless Application Protocol (WAP)

Inicialmente, é necessário esclarecer que o WAP possui duas versões em funcionamento no mercado, o WAP 1.0 e o WAP 2.0. A principal diferença entre as duas versões é que o WAP 1.0 lê apenas arquivos WML e o WAP 2.0 lê arquivos WML e HTML. Devido ao número reduzido atualmente de aparelhos que permitem o uso do WAP 2.0, todo o desenvolvimento do software será realizado com base no WAP 1.0, portanto nos basearemos na linguagem WML.

O WAP (Wireless Application Protocol) representa um link entre o desenvolvimento das comunicações móveis e o desenvolvimento do campo da transmissão de dados, particularmente a Internet. Usando Scripts WML (Wireless Markup Language) podemos programar aplicações a fim de apresentar informações no display do telefone após o estabelecimento de uma conexão WAP.

Para navegar pela internet através do celular é necessário se certificar que o mesmo possui um recurso chamado mini-browser. Praticamente todos os celulares presentes no mercado brasileiro atualmente possuem um mini-browser instalado. Os WAP browsers acessam páginas definidas em WML (Wireless Markup Language).

Para oferecer páginas WML acessíveis por celulares, precisamos somente de um servidor web, visto que a requisição da página é feita através dos protocolos TCP/IP e HTTP a partir do gateway da operadora e não pelo protocolo WAP como era de se esperar uma vez que o ponto originador de conexão é um aparelho celular. Este gateway irá receber o código WML da página e então encaminhá-lo para o cliente/celular.

5.4.2. Wireless Markup Language (WML)

WML é uma linguagem de programação baseada no XML (eXtensible Markup Language). A estrutura do WML é semelhante a encontrada no HTML para a confecção de páginas acessíveis por meio da internet. A especificação oficial do WML foi desenvolvida e é mantida pelo WAP Forum, um consórcio industrial fundado pela Nokia, Unwired Planet, Motorola, e Ericsson.

As especificações definem a sintaxe, variáveis e elementos usados no WML. Durante o desenvolvimento da WML o tamanho da página foi radicalmente reduzido desde o início e as próprias páginas foram modularizadas. Para isto, divide-se uma página em cards, um card individual pode ser exibido sem problemas em um display de telefone móvel. Vários de tais cards são combinados em um deck. Um deck é identificado via sua própria URL. Uma ilustração dos cards e decks pode ser encontrada abaixo:



Figura 6: Wireless Markup Language

Somente um card é exibido por vez na tela do aparelho celular, mas uma página não necessariamente precisa ter somente um card. Esse recurso surgiu devido ao fato de termos um limite muito pequeno de informação que pode ser exibida na tela do aparelho celular de uma só vez.

As tags delimitam o corpo do documento. Num WML podemos ter vários cards que podem ter referências entre si. Para a disponibilização das imagens para visualização, é necessário que o arquivo que comporta a imagem tenha um formato especial denominado wbmp ou que o celular suporte imagens do tipo gif, como é o caso dos celulares que utilizamos em nosso projeto. Portanto tomaremos como base que o médico utilizará um celular com suporte a WAP e que visualize imagens do tipo gif.

6. Programas utilizados

6.1. Desenvolvedor em PHP

Para o desenvolvimento do software em PHP, utilizamos o programa UltraEdit, o qual permite editar código em uma infinidade de linguagens de programação como HTML, Javascript, ASP, entre outras.

UltraEdit é uma ferramenta que substitui perfeitamente o Notepad. Você pode editar vários arquivos de uma só vez e de tamanho ilimitado e permite fazer coisas como o "substituir" de um texto em todo um conjunto de arquivos.

Entende Java, ASPs, JSPs, HTML e faz o syntax highlight de palavras chave. Inclusive tem a possibilidade de entender a linguagem PL/SQL de Oracle. Dispõem de macros e muito mais.

Permite editar em Hexadecimal e também acrescentar diferentes dicionários.

Também tem integrado o suporte para fazer FTP ao servidor onde tiver as páginas. Como se pode ver é um programa bastante completo e muito útil para os programadores de páginas web.

Pode-se conseguir uma versão shareware deste programa na própria web do produto www.ultraedit.com.

6.2. Apache

Para permitir que o PC que utilizamos no desenvolvimento do projeto seja acessado remotamente, tanto pelo celular como pela internet, é necessário que ele se torne um Servidor Web. Para tanto, necessitamos de um programa que atribua a ele essa função. O programa mais utilizado no mundo para servidores web é o Apache. Segundo dados da Netcraft Web Server em fevereiro de 2005, mais de 68% dos sites da internet usavam o Apache como servidor.

O Apache foi originalmente elaborado para permitir o acesso a páginas http(web), porém, com a evolução do WAP, ele também passou a poder ser utilizado para páginas em WML e não apenas HTML. Porém, há a necessidade de realizarmos algumas configurações no programa.

6.2.1. Configuração do Apache

Por padrão o Apache já vem habilitado para servir páginas WML. A única alteração a ser feita nesse caso é nos MIME types (Multipurpose Internet Mail Extensions) do servidor para que ele passe a reconhecer o WML, dizendo que os arquivos com extensão *.wml deverão retornar um cabeçalho do tipo Content-Type: text/vnd.wap.wml para a aplicação que fez a requisição.

Para tanto, deve-se analisar o arquivo mime.types (a localização pode variar de acordo com a versão do Apache) e certificar-se que existem as seguintes linhas:

application/vnd.wap.sic

```
application/vnd.wap.slc  
application/vnd.wap.wbxm1 wbxm1  
application/vnd.wap.wmlc wmlc  
application/vnd.wap.wmlscriptc wmlsc
```

Por fim, adicione as seguintes linhas no arquivo httpd.conf:

```
AddType text/vnd.wap.wml wml  
AddType image/vnd.wap.wbmp wbmp  
AddType text/vnd.wap.wmlscript wmls  
AddType application/vnd.wap.wmlc wmlc
```

Também deve ser configurado o diretório onde os documentos estarão, através da opção DocumentRoot. Basta procurar a opção no arquivo "httpd.conf" e escrever o nome do diretório em seguida, como no exemplo:

```
DocumentRoot "C:\VREC\"
```

Utilizamos VREC como a sigla para Visualizador Remoto de ECG via Celular

6.3. Desenvolvedor em WML

O código-fonte pode ser elaborado em qualquer editor de texto como por exemplo, o bloco de notas. Porém existem alguns kits de desenvolvimento para facilitar a programação e simulação dos códigos desenvolvidos. Dentre as várias opções disponíveis, escolhemos o Openwave Mobile [6], que permitiu simularmos os softwares desenvolvidos em WML através de um simulador de mini-browser de celular e assim não gastarmos quantias consideráveis de tráfego da rede da operadora de telefonia móvel para realizarmos os testes de operação. O simulador oferece um telefone padrão na tela do computador no qual podemos simular o tamanho da tela e o pressionamento de teclas de um aparelho celular e assim construir uma página compatível. Abaixo segue uma figura, apresentando o formato do simulador de celular:



Figura 7 –Simulador de Celular da Openwave

7. Descrição Funcional

Para melhor entendimento, esse item foi dividido em duas partes logicamente separáveis: O Software do PC e o Software do Servidor.

7.1. Projeto do Software do PC

O Software do PC tem o objetivo de interpretar os sinais vindos do hardware, analisar as formas de onda de ECG do sinal, e enviá-las para a interface de comunicação que, por conseguinte, encaminhará essa informação para o servidor.

A linguagem utilizada para seu desenvolvimento foi o PHP, e o código-fonte do software desenvolvido pode ser encontrado anexo ao presente relatório.

7.1.1. Módulo da placa

A função desse módulo é de realizar a interface entre o computador e o hardware utilizado, no caso a placa Lynx 10/26. Nela temos uma biblioteca com algumas rotinas desta placa, que são: a rotina de escolha de frequência na qual a placa vai funcionar; a rotina de leitura dos valores de tensão recebidos nas 16 entradas da placa; e a rotina de escrita que joga certos valores de tensão nas 2 saídas dessa placa.

7.1.2. Módulo de captura

Esse módulo utiliza-se da parte do módulo da placa para comandar a placa Lynx, onde foi escolhida a frequência de 500 Hz e uma quantidade de 1000 amostras, ou seja, conseguimos ter os dados de 2s de um eletrocardiograma. Além disso, em comparação com o trabalho de formatura 2003 “Monitoração Remota de Sinais Vitais” [1], de onde tiramos idéias iniciais de HW, onde as derivações eram feitas no próprio HW e elas eram enviadas para o computador, agora o HW adquire direto os valores de tensões das partes do corpo humano (braços esquerdo e direito, e pernas esquerda e direita) e as derivações são feitas no próprio SW, como pode ser visto no código anexo.

Para essa parte, temos um arquivo em C que além de comandar a placa, já gera o programa em PHP, onde que o mesmo será explicado em detalhes logo abaixo.

7.1.3. Módulo de geração da imagem

Dentro da linguagem PHP, temos várias classes de PHP, que são um conjunto de funções de programação, com algumas funções úteis para o desenvolvimento da parte do software responsável pela geração da imagem. Com relação à classe encontrada, temos umas funções que foram adaptadas, outras que foram adicionadas, e as que não

tiveram alteração por não terem utilidade para o projeto ou por já estarem adequadas, sendo assim, temos as funções:

- que ativa o construtor de imagens
- que "destrói" a imagem
- que mostra a imagem na tela e/ou cria um arquivo de imagem png ou jpg
- que cria um array associado aos valores de cores
- que colore a célula com uma determinada cor
- que redimensiona a imagem para encaixar na dimensão escolhida
- que calcula o tamanho da imagem resultante

Sendo assim, em anexo temos um exemplo de um dos arquivos PHP que são gerados, pois todos são muito parecidos, onde a única coisa que muda entre eles são os dados, e com isso a imagem muda. Esse arquivo abre uma biblioteca de classes PHP que são utilizadas por ele, depois temos os dados que gerarão a figura (que devem obrigatoriamente ter um ponto para separação decimal e uma vírgula para separação entre eles), assim como as 1000 amostras que estão ligadas cada uma a um dos dados. Depois escolhemos o tamanho total da figura (640x480 para a figura do computador e 150x150 para a figura do celular), ao tamanho dessa área que iremos utilizar para desenhar e em que ponto começamos o desenho (590x430 e (30,20) para a figura do computador e 130x130 e (10,10) para a figura do celular), ligamos os pontos dos dados com uma reta, dimensionamos a escala y do gráfico, assim como alguns traços, e finalmente escolhemos o tipo de figura que será gerada, que no caso foi escolhida figuras em GIF.

7.2. Projeto do Software do Servidor

O Software do Servidor tem o objetivo de receber as informações do PC da residência e fornecer a plataforma necessária para que o médico possa definir o nome do paciente a ser consultado e visualizar as formas de onda através de seu celular via interface WAP.

7.2.1. Especificações do software

A comunicação do PC para o servidor é unidirecional, no sentido de que o Servidor apenas recebe informações do PC da residência. A informação recebida é a figura de ECG no formato gif, que foi gerada pelo software em PHP do PC da residência.

7.2.2. A página WAP

A página WAP foi desenvolvida em linguagem WML. Conforme mencionado anteriormente, a linguagem especifica a divisão da página em cards e decks. Para a escolha do paciente e visualização das imagens projetamos os cenários de páginas a seguir. Para melhor entendimento de como as páginas são visualizadas pelo médico, apresentaremos figuras de telas de celular obtidas com o simulador de celular da Openwave[6].

7.2.2.1. Cenário I

7.2.2.1.1. Deck 1 - Apresentação

Este deck possibilita a escolha dentre os nomes de pacientes que foram analisados e que consequentemente possuem um arquivo com a imagem do eletrocardiograma gravado no HD. Conforme figura abaixo, pode-se observar a tela que o médico visualiza em seu celular. Nesse card o médico pode percorrer pelas tags com os nomes dos pacientes e selecionar aquele o qual deseja observar a respectiva figura de ECG. O card apresenta também uma outra tag com a opção de se abrir um outro card o qual são apresentadas as instruções de utilização da página.



Figura 8 – Card “Escolha de paciente”



Figura 9 – Card “Instruções”

7.2.2.1.2. Deck 2 - Figuras de ECG

Após ter sido clicado o nome do paciente no deck de apresentação, é aberto o deck de Figura de ECG do respectivo paciente. No caso da figura abaixo foi aberta a figura do paciente *Edson Mota*. Este deck possui apenas um card e além da apresentação do ECG a única ação que pode ser feita é o retorno para a página anterior através do clique na tag denominada “Anterior”.

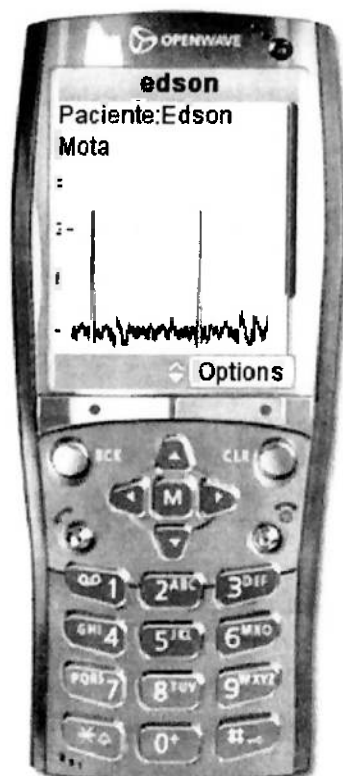


Figura 10 – Card “Figura de ECG” – Paciente *Edson Mota*

7.2.2.2. Cenário II

7.2.2.2.1. Deck 1- Apresentação:

Este deck apresenta apenas duas tags (“Mostrar Figuras” e “Instruções”). Ao clicar na tag “Mostrar Figuras” é aberto um outro deck com a figura de ECG do primeiro paciente que tem figura na pasta de imagens do HD. Ao clicar na tag “Instruções” são apresentadas as instruções de utilização da página.



Figura 11 – Card “Escolha de paciente”

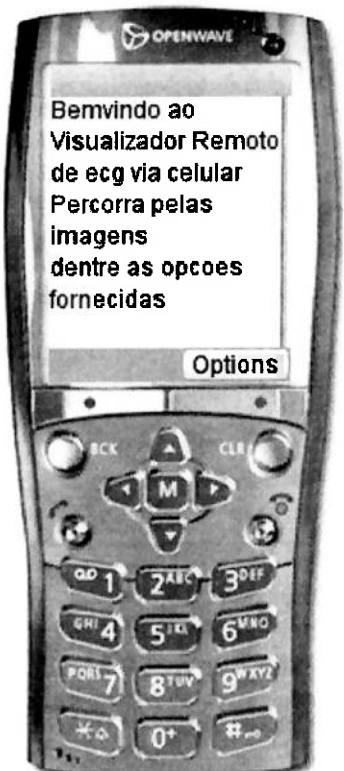


Figura 12 – Card “Instruções”

7.2.2.2.2. Deck 2 - Figuras de ECG

Após ter sido clicado na tag “Mostrar Figuras” esse deck é aberto no qual pode-se visualizar o ECG do primeiro paciente que tem figura na pasta de imagens do HD. Diferentemente do que ocorre no deck elaborado para o cenário 1, esse card apresenta duas tags (“Próxima” e “Anterior”) onde pode-se percorrer pelos ECG dos outros pacientes.



Figura 13 – Card “Figura de ECG” – Paciente *Felipe*

8. Testes e Validação

8.1. Aquisição dos dados

Para testar a aquisição dos dados realizamos os seguintes procedimentos:

- 1) Ajustamos o simulador de ECG para gerar um sinal com as seguintes características:
 - a. Uma pessoa normal com frequência cardíaca de 80 bpm (código 168)
 - b. Um recém nascido com frequência cardíaca de 200 bpm (código 172)
 - c. Uma arritmia atrial grossa no eletrocardiograma (código 12)
- 2) Captamos o sinal do simulador com a ajuda de um osciloscópio e geramos o arquivo de dados com a ferramenta do próprio osciloscópio, onde os dados foram tratados de forma a mostrarem todas as 6 derivações de 1.
- 3) Captamos o sinal do simulador com as mesmas características utilizando um computador equipado com a placa Lynx e os softwares de controle da placa cujos códigos-fonte estão anexados a este relatório.

As respostas dos dois arquivos estão apresentadas abaixo:

OBS: Apresentamos apenas as respostas obtidas pela derivação entre o braço esquerdo e a perna esquerda, pois a única coisa que varia entre eles são os valores dos dados. Não apresentamos todos os valores, pois não faz sentido termos mais 10 páginas somente com dados numéricos.

Para o arquivo do osciloscópio, temos:

-0.0625
-0.04688
-0.09375
0.015625
-0.09375
-0.14063
-0.10938
-0.09375
0
-0.125
-0.09375
-0.09375
-0.01563
-0.07813
0.046875
-0.09375
0.015625
-0.10938
-0.09375
-0.03125
-0.04688
-0.0625

```

0.03125
...
...
...
0.015625
-0.20313
-0.09375
-0.07813
-0.09375

```

Onde devemos ressaltar que foi utilizado o Excel do Office 2003, e nele foi escolhida a opção de se trocar o separador decimal de vírgula para ponto, que é como deve ser tratado os dados, entretanto se for utilizado outra versão do Excel, essa opção pode não existir, e sendo assim teria de ser utilizado uma combinação de programas, como o Word para ajudar no tratamento dos dados para o PHP, além disso, pelos dois métodos, é necessária uma vírgula após cada valor de dado, e isso é facilmente conseguido colocando-se uma vírgula no Excel e utilizando a função CONCATENAR. Após os dados estarem prontos é só copiá-los na parte de dados (que é a única que se altera) do arquivo padrão de PHP.

Já para o arquivo gerado pelo programa em C, utilizando a placa Lynx, temos:

```

<?php
require_once("../lib/phpchartdir.php");

$data = array(-0.0625,-0.046875,-0.09375,0.015625,-0.09375,-0.140625,-0.109375,-
0.09375,0,-0.125,-0.09375,-0.09375,-0.015625,-0.078125,0.046875,-0.09375,0.015625,-
0.109375,-0.09375,-0.03125,-0.046875,-0.0625,0.03125,-0.078125,-0.09375,-0.09375,-
0.046875,-0.09375,0.03125,0.03125,-0.078125,-0.03125,0.015625,-0.015625,-
0.0625,0.03125,-0.109375,-0.078125,0.015625,-0.046875,-0.015625,-0.015625,0,0,-
0.125,-0.0625,0.03125,-0.125,-0.09375,-0.09375,-0.140625,-0.078125,-0.109375,-
0.09375,-0.125,-0.0625,-0.078125,-0.03125,-0.03125,-0.0625,-0.078125,-0.015625,-
0.078125,-0.09375,-0.03125,-0.125,-0.046875,-0.046875,-0.0625,-0.0625,-0.046875,-
0.015625,-0.0625,-0.03125,-0.0625,-0.046875,0,-0.015625,0.046875,-0.0625,-
0.078125,-0.078125,-0.046875,-0.015625,0.046875,-0.046875,-0.15625,-0.015625,-
0.125,-0.296875,-0.296875,-0.296875,-0.265625,-0.53125,-0.625,-0.65625,-0.734375,-
0.65625,-0.484375,-0.578125,-0.359375,-0.453125,-0.203125,-0.25,-0.15625,-
0.109375,-0.03125,0.078125,-0.09375,-0.015625,-0.015625,-0.109375,-0.015625,-
0.125,0.015625,0.0625,0.03125,-0.015625,-0.015625,-0.109375,0,-0.09375,-0.046875,-
0.015625,-0.046875,0.03125,0.03125,0.015625,-0.078125,-0.046875,-0.109375,-
0.03125,-0.03125,-0.046875,-0.109375,-0.0625,-0.0625,-0.046875,0.03125,-0.109375,-
0.015625,-0.015625,-0.125,-0.03125,-0.046875,0,-0.171875,-0.015625,-0.046875,-
0.078125,-0.109375,-0.09375,-0.03125,0.046875,-0.078125,-0.109375,-0.09375,-
0.1875,-0.0625,-0.09375,-0.078125,-0.015625,0,-0.046875,-0.09375,-0.03125,-0.125,-
0.109375,-0.015625,-0.09375,-0.03125,-0.015625,0,0.015625,-0.0625,-0.09375,-
0.046875,-0.046875,-0.0625,-0.015625,-0.109375,0.03125,-0.015625,-
0.015625,0.03125,-0.03125,-0.0625,-0.078125,-0.078125,-0.015625,-0.078125,-
0.140625,-0.078125,-0.15625,-0.0625,-0.140625,-0.15625,-0.046875,-0.0625,-
0.109375,-0.078125,-0.0625,-0.109375,-0.234375,-0.203125,-0.15625,-0.1875,-
0.203125,-0.078125,-0.125,-0.25,-0.203125,-0.203125,-0.1875,-0.21875,-0.125,-

```

0.1875,-0.125,-0.15625,-0.15625,-0.09375,-0.140625,-0.203125,-0.1875,-0.015625,
 -0.140625,-0.140625,-0.140625,-0.1875,-0.15625,-0.140625,-0.125,-0.109375,-
 0.109375,-0.109375,-0.125,-0.125,-0.28125,-0.203125,-0.15625,-0.09375,-0.1875,-
 0.078125,-0.140625,0.046875,-0.171875,-0.015625,-0.125,-0.09375,-
 0.03125,0.015625,-0.03125,-0.015625,0,-0.09375,-0.078125,-0.09375,0.0625,-
 0.03125,-0.046875,-0.0625,-0.15625,-0.015625,-0.03125,-0.078125,-0.078125,-
 0.09375,0.03125,-0.0625,-0.0625,-0.046875,0,-0.078125,0.015625,-0.125,0,0.03125,-
 0.078125,-0.078125,-0.015625,-0.15625,-0.03125,-0.09375,0,0.015625,0.03125,-
 0.140625,-0.046875,-0.015625,0,-0.015625,-0.046875,-0.0625,-0.03125,0.015625,-
 0.03125,-0.0625,-0.09375,0.0625,-0.015625,-0.09375,-0.03125,-0.1875,0.046875,-
 0.03125,0.078125,-0.109375,-0.09375,-0.140625,-0.0625,-0.046875,-0.0625,0,-
 0.015625,-0.0625,-0.03125,-0.03125,-0.078125,-0.15625,-0.0625,-0.0625,-0.015625,-
 0.078125,0.03125,-0.09375,-0.0625,-0.0625,-0.046875,-0.09375,0.09375,0.0625,-
 0.015625,-0.078125,0,0.015625,0.046875,-0.03125,0,0,-0.015625,-0.125,0.0625,-
 0.046875,-0.15625,-0.0625,-0.046875,-0.109375,-0.09375,-0.046875,-0.140625,-
 0.09375,-0.09375,-0.109375,0.09375,0,-0.015625,-0.09375,-0.046875,-0.078125,-
 0.0625,-0.125,-0.09375,-0.125,-0.03125,-0.015625,0.015625,0.015625,-0.078125,-
 0.046875,-0.046875,0.03125,0,0,-0.09375,-0.078125,-0.046875,-0.0625,-0.09375,-
 0.0625,0,-0.09375,-0.09375,-0.015625,-0.03125,-0.0625,-0.203125,-0.234375,-
 0.109375,-0.15625,-0.09375,-0.078125,-0.046875,-0.125,-0.09375,-0.078125,0,-
 0.046875,-0.140625,-0.140625,-0.0625,-0.078125,-0.09375,-0.078125,-0.09375,-
 0.09375,0,0,-0.015625,-0.078125,-0.03125,-0.03125,-0.03125,0.015625,-0.078125,-
 0.015625,-0.109375,-0.03125,-0.09375,-0.03125,-0.09375,0.015625,-0.140625,-
 0.046875,-0.015625,-0.125,-0.09375,-0.109375,-0.09375,-0.03125,-0.0625,-0.0625,-
 0.03125,-0.03125,-0.046875,-0.09375,-0.046875,-0.0625,-0.15625,-0.078125,-
 0.078125,-0.03125,-0.046875,-0.0625,0.015625,-0.046875,-0.0625,0,0,-0.015625,-
 0.015625,-0.015625,-0.046875,-0.0625,-0.0625,0,-0.125,-0.125,-0.125,0.0625,-
 0.03125,-0.03125,0.03125,-0.171875,-0.140625,-0.15625,-0.21875,-0.21875,-0.53125,-
 0.484375,-0.625,-0.625,-0.71875,-0.53125,-0.578125,-0.421875,-0.515625,-0.3125,-
 0.21875,-0.21875,-0.21875,-0.0625,0.0625,-0.0625,0.03125,-0.03125,-0.046875,-
 0.0625,-0.046875,-0.03125,0,-0.046875,0.078125,-0.03125,0.03125,-0.046875,
 0.03125,0.015625,0.0625,-0.0625,0,-0.0625,-0.015625,0.015625,-0.09375,-0.078125,-
 0.0625,-0.0625,0.015625,-0.125,-0.0625,-0.046875,-0.03125,-0.0625,-
 0.078125,0.015625,-0.109375,0.015625,-0.03125,0.03125,-0.09375,-0.078125,-
 0.03125,-0.09375,-0.03125,-0.015625,0,-0.109375,0.0625,-0.03125,-0.140625,0,-
 0.109375,-0.0625,0.015625,-0.0625,-0.015625,-0.046875,-0.046875,0.015625,0,-
 0.03125,-0.078125,-0.046875,0,-0.109375,-0.046875,-0.078125,-0.125,0,0.046875,-
 0.0625,-0.046875,-0.046875,-0.03125,-0.03125,0.015625,-0.03125,0.0625,-0.078125,-
 0.078125,0.015625,0,0.03125,-0.046875,-0.03125,-0.046875,0.015625,-
 0.078125,0,0.015625,-0.078125,-0.109375,-0.125,-0.171875,-0.171875,-0.140625,-
 0.140625,-0.140625,-0.15625,-0.09375,-0.203125,-0.1875,-0.21875,-0.109375,-
 0.140625,-0.09375,-0.171875,-0.15625,-0.140625,-0.125,-0.171875,-0.125,-0.171875,-
 0.171875,-0.171875,-0.09375,-0.15625,-0.1875,-0.03125,-0.140625,-0.078125,-
 0.171875,-0.109375,-0.171875,-0.09375,-0.15625,-0.140625,-0.109375,-0.09375,-
 0.046875,-0.125,-0.21875,-0.171875,-0.0625,-0.09375,-0.15625,-0.140625,-0.109375,-
 0.109375,-0.140625,-0.046875,-0.078125,0.046875,-0.046875,-0.015625,-0.09375,-
 0.078125,-0.0625,-0.09375,-0.015625,-0.0625,-0.046875,-0.09375,-0.046875,-
 0.03125,-0.078125,-0.015625,-0.09375,-0.046875,-0.078125,-0.0625,-
 0.078125,0.046875,-0.0625,-0.03125,0.03125,0.046875,0,-0.109375,-0.078125,-
 0.109375,-0.03125,-0.03125,0,0.015625,0.03125,0.03125,-0.0625,-0.046875,-

```

0.015625,0.0625,-0.09375,-0.046875,-0.03125,-0.109375,0.03125,
0.0625,0.109375,0,-0.0625,0.046875,-0.09375,0,-0.09375,-0.125,-0.03125,-0.09375,-
0.0625,0.03125,0,-0.03125,-0.078125,0,-0.0625,-0.078125,-0.046875,0.046875,-
0.171875,0.03125,0.03125,-0.015625,-0.015625,-0.09375,0.03125,-0.046875,-0.0625,-
0.09375,-0.109375,-0.0625,0.03125,-0.046875,-0.078125,-0.046875,-
0.046875,0.015625,-0.0625,-0.140625,-0.109375,0,-0.078125,0,-0.109375,-0.03125,
-0.09375,-0.078125,-0.03125,0.0625,-0.078125,0,-0.03125,-0.0625,-0.015625,-
0.078125,-0.015625,0.03125,-0.046875,-0.078125,-0.09375,-0.015625,-0.109375,-
0.109375,-0.078125,-0.0625,-0.078125,-0.046875,-0.015625,-0.03125,0,-0.078125,-
0.125,-0.109375,0.0625,-0.015625,0.046875,0.03125,-0.09375,-0.109375,0.03125,-
0.125,-0.0625,-0.078125,-0.078125,0.03125,-0.125,0.03125,0.015625,-0.140625,-
0.078125,-0.09375,-0.203125,-0.0625,-0.171875,-0.109375,-0.125,-0.125,0,0,-
0.03125,-0.15625,-0.140625,-0.109375,-0.03125,-0.03125,-0.15625,-0.0625,-
0.015625,-0.015625,-0.015625,-0.0625,-0.046875,-0.046875,-0.125,-0.03125,-
0.015625,-0.0625,-0.015625,-0.046875,-0.015625,0.046875,0.0625,-0.046875,-0.0625,-
0.015625,-0.0625,0.015625,-0.09375,-0.046875,-0.078125,-0.125,-0.09375,-0.03125,-
0.015625,-0.046875,-0.09375,-0.0625,-0.125,-0.046875,-0.046875,-0.0625,0.046875,0,-
0.125,-0.0625,-0.140625,-0.0625,-0.09375,-0.046875,-0.046875,-0.109375,-0.046875,-
0.015625,-0.171875,-0.109375,-0.09375,-0.15625,-0.09375,-0.078125,0,-
0.03125,0.03125,-0.203125,-0.046875,-0.140625,-0.046875,-0.046875,-0.15625,-
0.015625,-0.03125,-0.125,-0.21875,-0.359375,-0.46875,-0.34375,-0.421875,-0.59375,-
0.734375,-0.703125,-0.640625,-0.609375,-0.421875,-0.484375,-0.421875,-0.234375,-
0.21875,-0.109375,-0.109375,-0.0625,0.03125,-0.0625,-0.09375,0.109375,-0.078125,-
0.015625,-0.03125,-0.0625,0.015625,-0.046875,-0.046875,-0.078125,-0.0625,-
0.03125,-0.03125,-0.0625,-0.046875,-0.03125,0.03125,-0.03125,-0.046875,-0.109375,-
0.015625,-0.015625,-0.0625,0.015625,0.015625,-0.09375,-0.125,-
0.140625,0.03125,0.015625,-0.109375,-0.046875,-0.078125,-0.0625,0.078125,-
0.0625,-0.046875,-0.109375,-0.046875,-0.015625,-0.125,-0.1875,-0.1875,0.015625,-
0.09375,0.046875,-0.078125,-0.0625,-0.015625,0.015625,-0.015625,-0.078125,-
0.03125,-0.046875,0,-0.015625,-0.0625,0.03125,0.109375,-0.015625,-0.09375,-
0.125,0,-0.046875,-0.09375,0.03125,-0.046875,0.03125,-0.125,-0.09375,-0.0625,-
0.09375,-0.015625,-0.03125,-0.125,-0.109375,-0.125,0,-0.140625,-0.015625,0.015625,-
0.046875,-0.0625,-0.1875,-0.09375,0,-0.078125,-0.078125,-0.046875,-0.078125,-
0.21875,-0.03125,-0.234375,-0.125,-0.171875,-0.203125,-0.046875,-0.15625,-0.0625,-
0.21875,-0.140625,-0.21875,-0.203125,-0.171875,-0.078125,-0.09375,-0.265625,-
0.171875,-0.140625,-0.171875,-0.09375,-0.15625,-0.203125,-0.21875,-0.109375,-
0.0625,-0.078125,-0.171875,-0.1875,-0.15625,-0.140625,-0.09375,-0.171875,-
0.171875,-0.109375,-0.078125,-0.109375,-0.21875,-0.171875,-0.15625,-0.125,-
0.265625,-0.03125,-0.125,-0.078125,-0.109375,0.015625,-0.203125,-0.09375,-
0.078125,-0.09375,
);

```

\$labels =

```

array(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,3
0,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,5
8,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,8
6,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,1
10,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,1
30,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,1
50,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,1

```

70,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,1
 90,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,2
 10,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,2
 30,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,2
 50,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,2
 70,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,2
 90,291,292,293,294,295,296,297,298,299,300,301,302,303,304,305,306,307,308,309,3
 10,311,312,313,314,315,316,317,318,319,320,321,322,323,324,325,326,327,328,329,3
 30,331,332,333,334,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,3
 50,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,3
 70,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,3
 90,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,408,409,4
 10,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,427,428,429,4
 30,431,432,433,434,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,4
 50,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,4
 70,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,4
 90,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,5
 10,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,5
 30,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,5
 50,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,5
 70,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,5
 90,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,6
 10,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,6
 30,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,6
 50,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,6
 70,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,6
 90,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,7
 10,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,7
 30,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,7
 50,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,768,769,7
 70,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,786,787,788,789,7
 90,791,792,793,794,795,796,797,798,799,800,801,802,803,804,805,806,807,808,809,8
 10,811,812,813,814,815,816,817,818,819,820,821,822,823,824,825,826,827,828,829,8
 30,831,832,833,834,835,836,837,838,839,840,841,842,843,844,845,846,847,848,849,8
 50,851,852,853,854,855,856,857,858,859,860,861,862,863,864,865,866,867,868,869,8
 70,871,872,873,874,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,8
 90,891,892,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,9
 10,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,9
 30,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,948,949,9
 50,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,966,967,968,969,9
 70,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,9
 90,991,992,993,994,995,996,997,998,999,);

```
$c = new XYChart(640, 480);
```

```
$c->setPlotArea(30, 20, 590, 430);
```

```
$c->addLineLayer($data);
```

```
$c->xAxis->setLabels($labels);
```

```
$c->xAxis->setLabelStep(3);

header("Content-type: image/gif");
print($c->makeChart2(GIF));
?>
```

Para o arquivo do computador e:

```
$c = new XYChart(150, 150);

$c->setPlotArea(10, 10, 130, 130);

$c->addLineLayer($data);

$c->xAxis->setLabels($labels);

$c->xAxis->setLabelStep(3);

header("Content-type: image/gif");
print($c->makeChart2(GIF));
?>
```

Para o arquivo do celular.

Comparando as duas respostas, podemos verificar que o arquivo gerado no arquivo em C já gera o PHP diretamente, já o arquivo gerado pelo osciloscópio precisa que os dados sejam transformados num formato padrão para poderem ser adicionados na parte de dados do arquivo em PHP. Desconsiderando essas diferenças de formatação de texto, podemos avaliar que o módulo de aquisição de dados se comporta conforme esperado.

8.2. Geração da figura

Para testar o software que gera a figura realizamos os seguintes procedimentos:

- 1) Pegamos os arquivos gerados no item 8.1.
- 2) Colocamos esses arquivos num computador com algumas configurações de servidor.
- 3) Abrimos o arquivo em PHP no próprio browser do Internet Explorer, que executa o arquivo em PHP e gera as figuras que queremos na tela.

As figuras obtidas pelos dois métodos estão apresentadas abaixo:

Primeiramente, temos a figura gerada pelo osciloscópio, depois a figura gerada para o computador e por final a figura gerada para o celular, assim, temos:

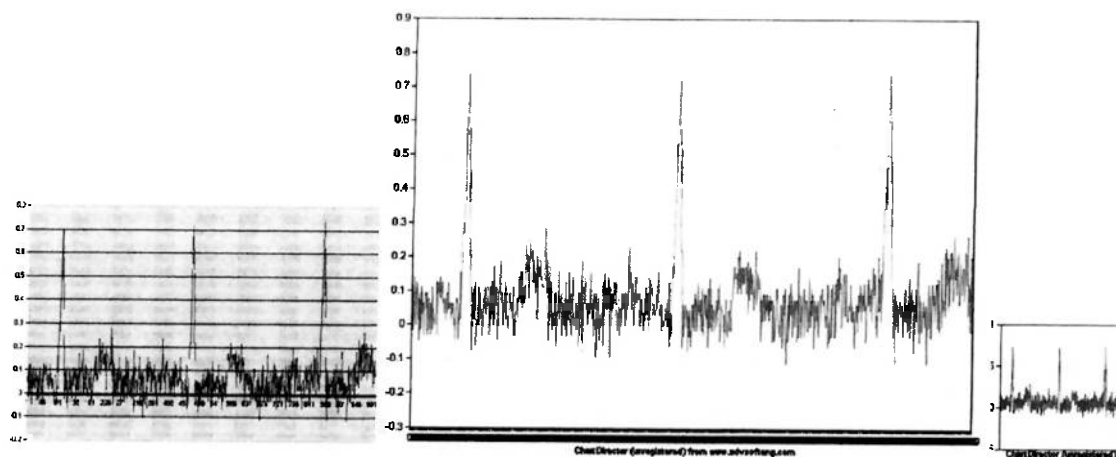


Figura 14 - Para a derivação entre braço esquerdo e perna esquerda à 80bpm

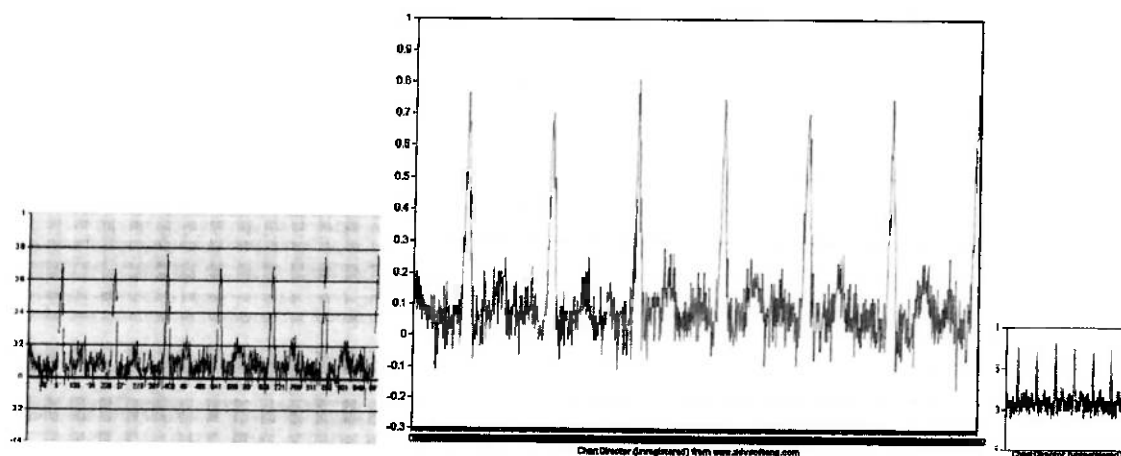


Figura 15 - Para a derivação entre braço esquerdo e perna esquerda à 200bpm

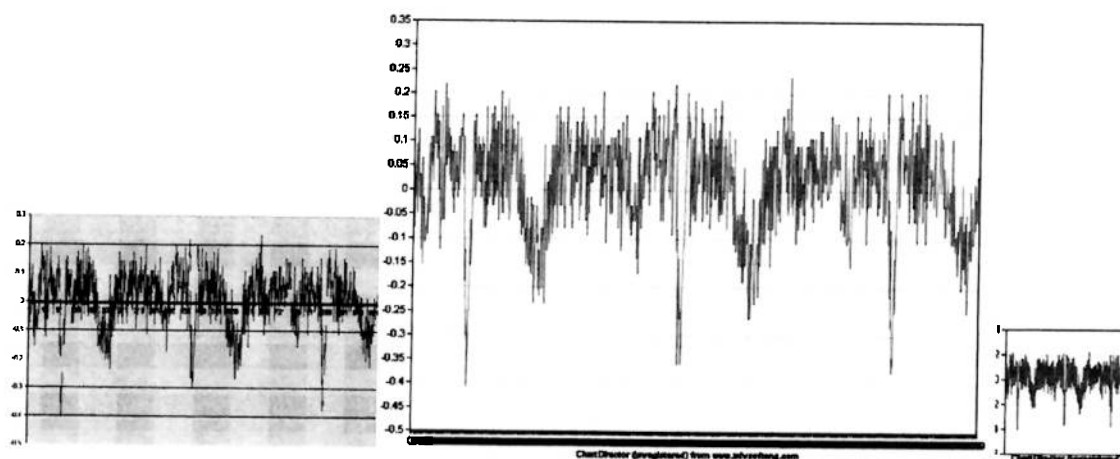


Figura 16 - Para a derivação entre braço esquerdo e perna direita à 80bpm

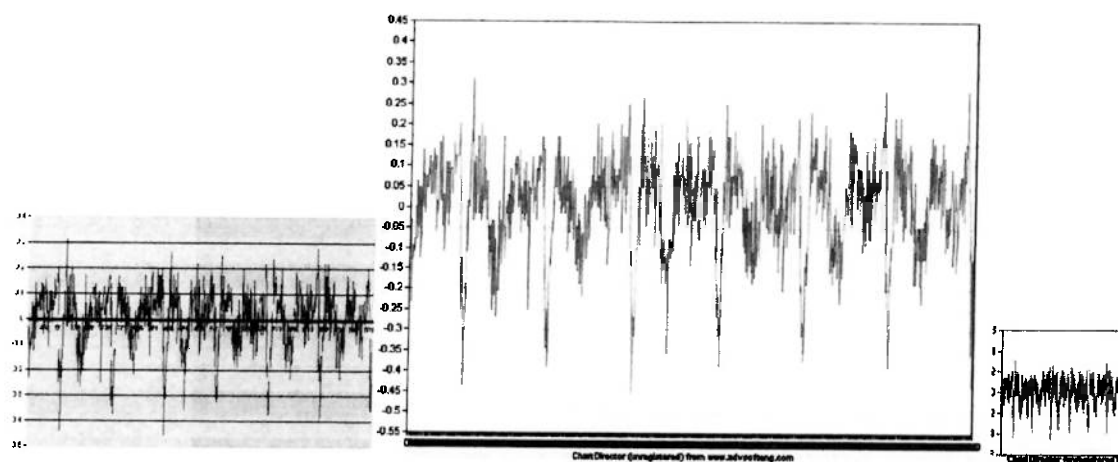


Figura 17 - Para a derivação entre braço esquerdo e perna direita à 200bpm

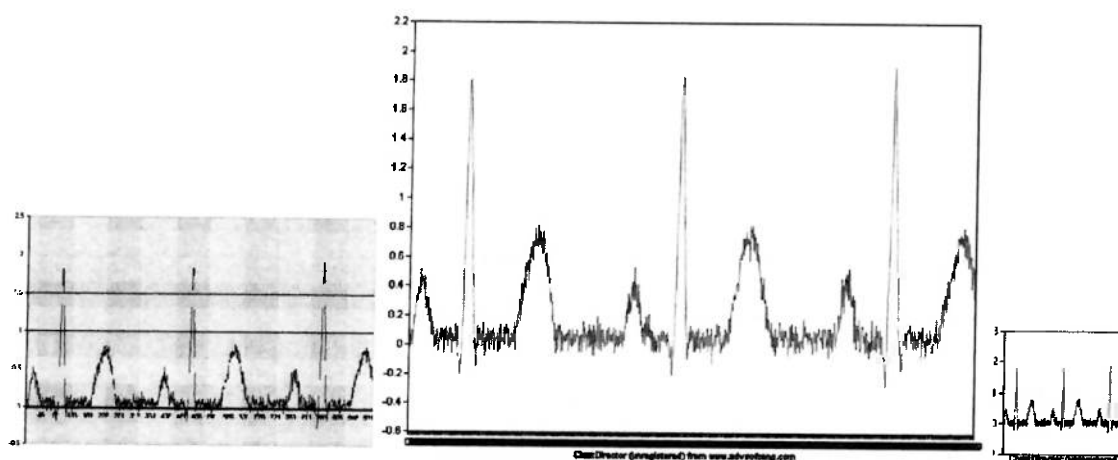


Figura 18 - Para a derivação entre braço direito e braço esquerdo à 80bpm

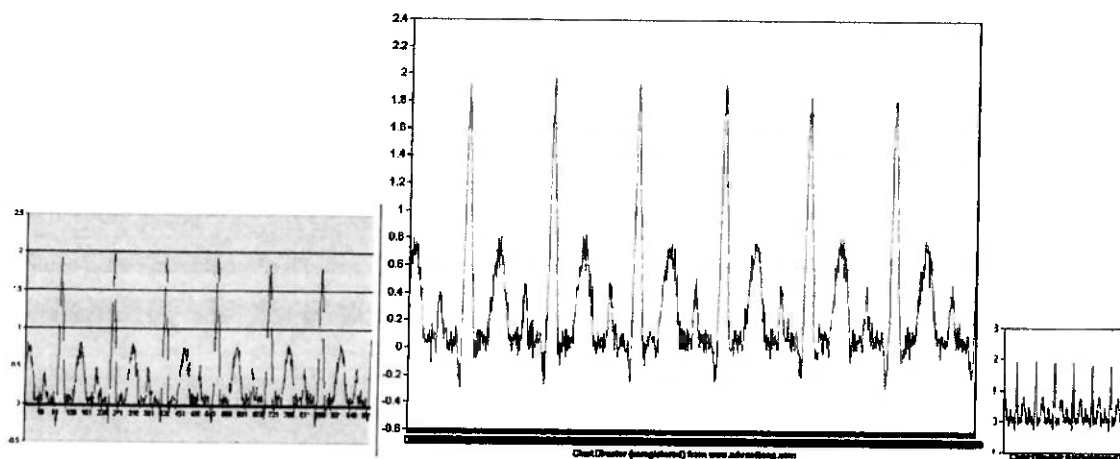


Figura 19 - Para a derivação entre braço direito e braço esquerdo à 200bpm

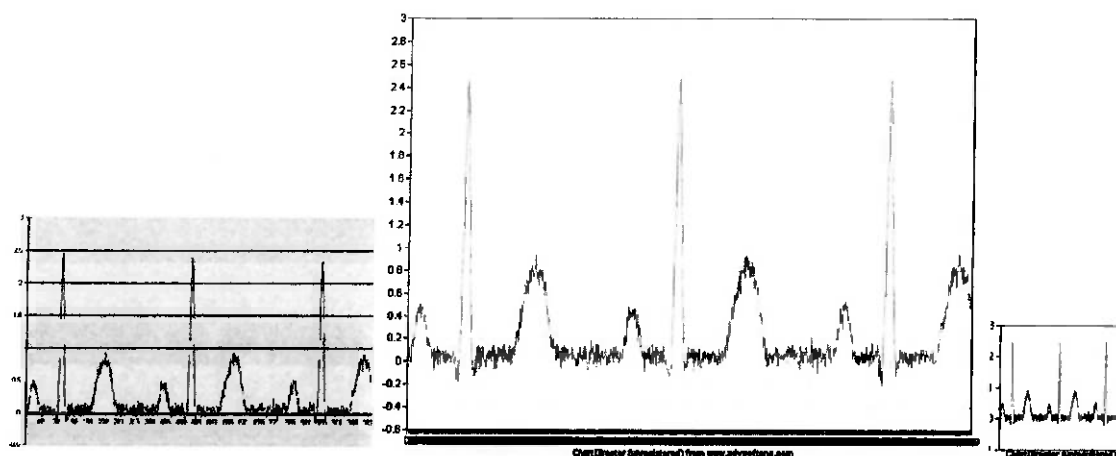


Figura 20 - Para a derivação entre braço direito e perna esquerda à 80bpm

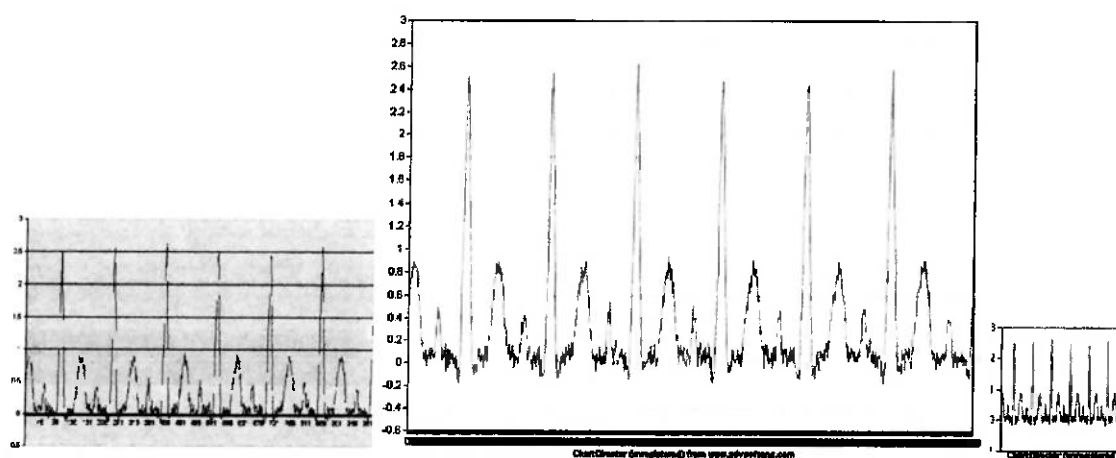


Figura 21 - Para a derivação entre braço direito e perna esquerda à 200bpm

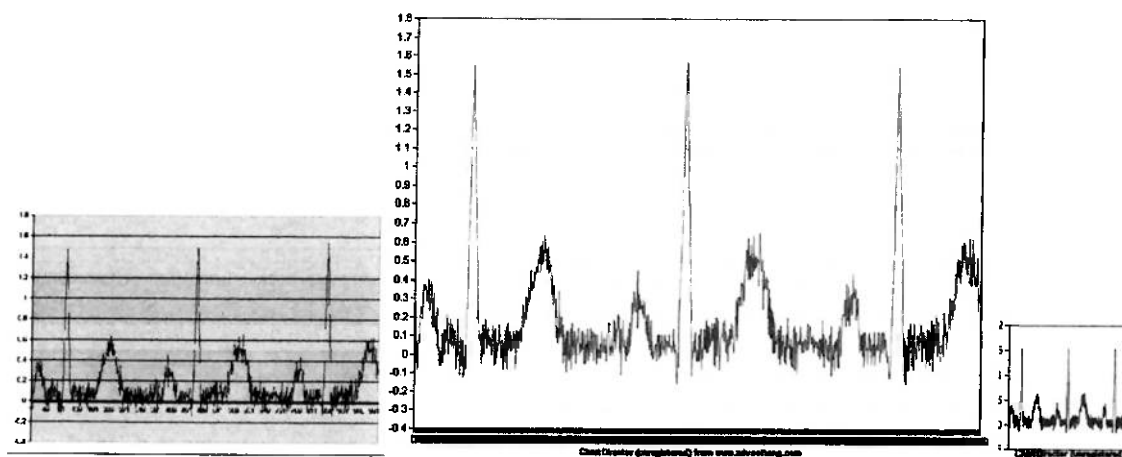


Figura 22 - Para a derivação entre braço direito e perna direita à 80bpm

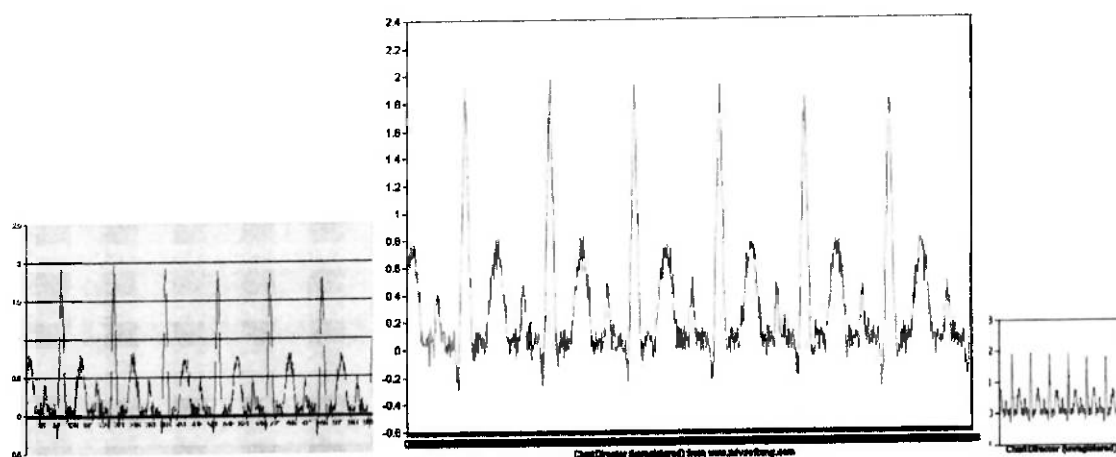


Figura 23 - Para a derivação entre braço direito e perna direita à 200bpm

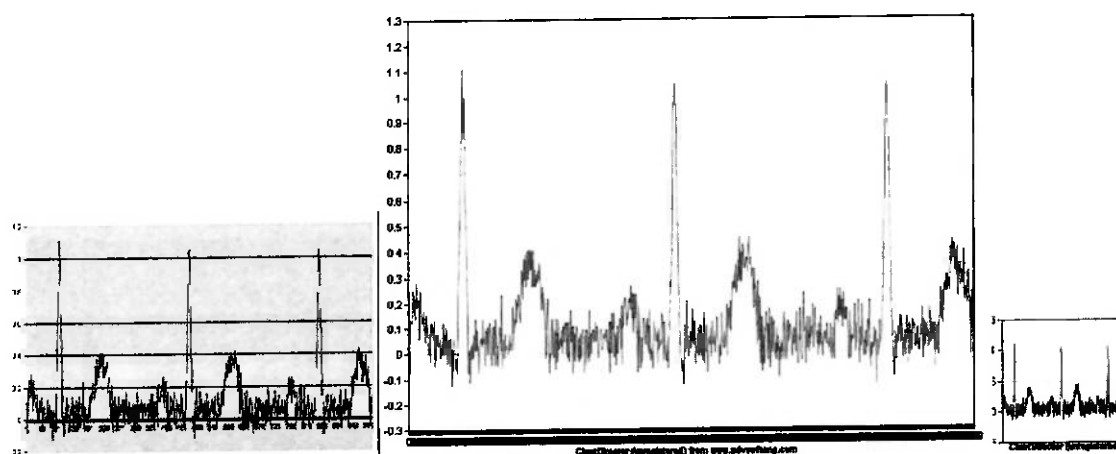


Figura 24 - Para a derivação entre perna direita e perna esquerda à 80bpm

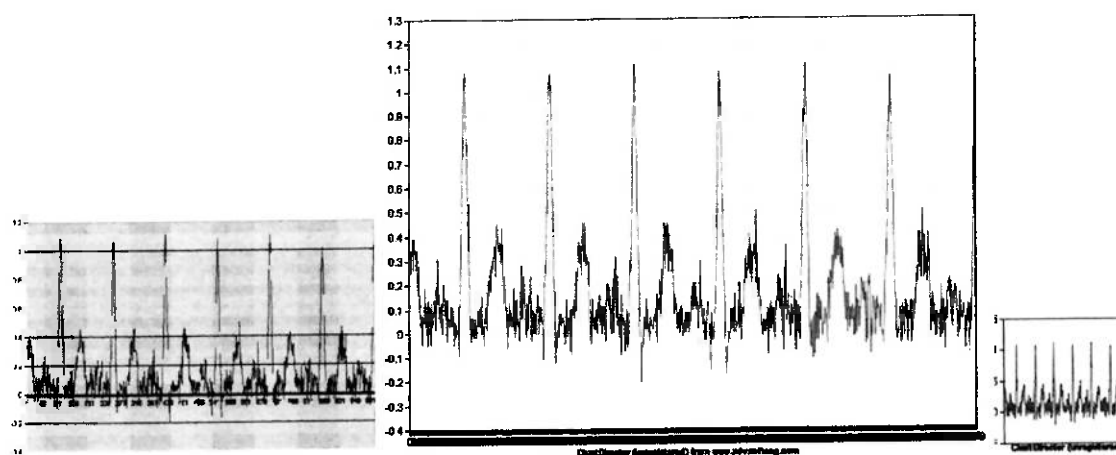


Figura 25 - Para a derivação entre perna direita e perna esquerda à 80bpm

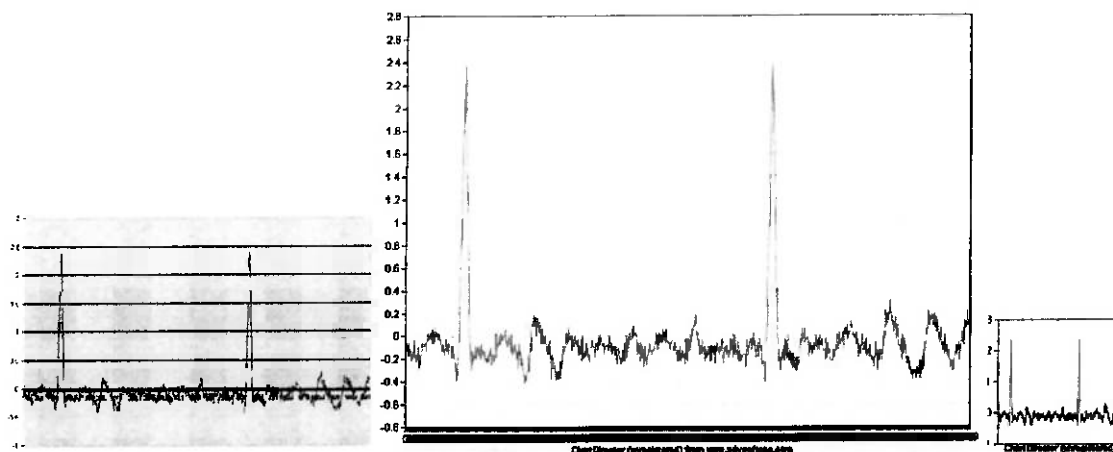


Figura 26 - Para a derivação entre braço direito e perna esquerda com arritmia atrial

Como podemos ver, as figuras geradas são muito parecidas, mostrando que elas mostram realmente como é o sinal do ECG adquirido, assim como é claro a diferença de resolução entre a figura que é vista no computador e a que é vista no celular. Assim, comprovamos o funcionamento do software do PC no que tange a geração da imagem a partir dos dados obtidos da placa Lynx.

8.3. Transferência e visualização da figura no celular e na página Web

Para testar a transferência da figura WAP e conseqüente visualização no celular, e para testar o acesso via Web pelo médico, realizamos os seguintes procedimentos:

- 1) Tomamos como base figuras geradas pelo software desenvolvido em PHP e as armazenamos no servidor do LEB, em <http://www.leb.usp.br/Jairo/final1/>.
- 2) Acessamos remotamente a página, simulando uma transação WAP, através do simulador da Openwave[6] e visualizamos a página de interesse como se fosse visualizada pelo celular.
- 3) Acessamos remotamente a página via Web, através do software Opera 8.51 [10], que é um browser como o Internet Explorer, só que a diferença é que ele suporta a abertura de páginas em WML, o que não ocorre com o Explorer.

Simulando as páginas desenvolvidas em WML nesses 2 softwares, obtivemos as seguintes imagens:



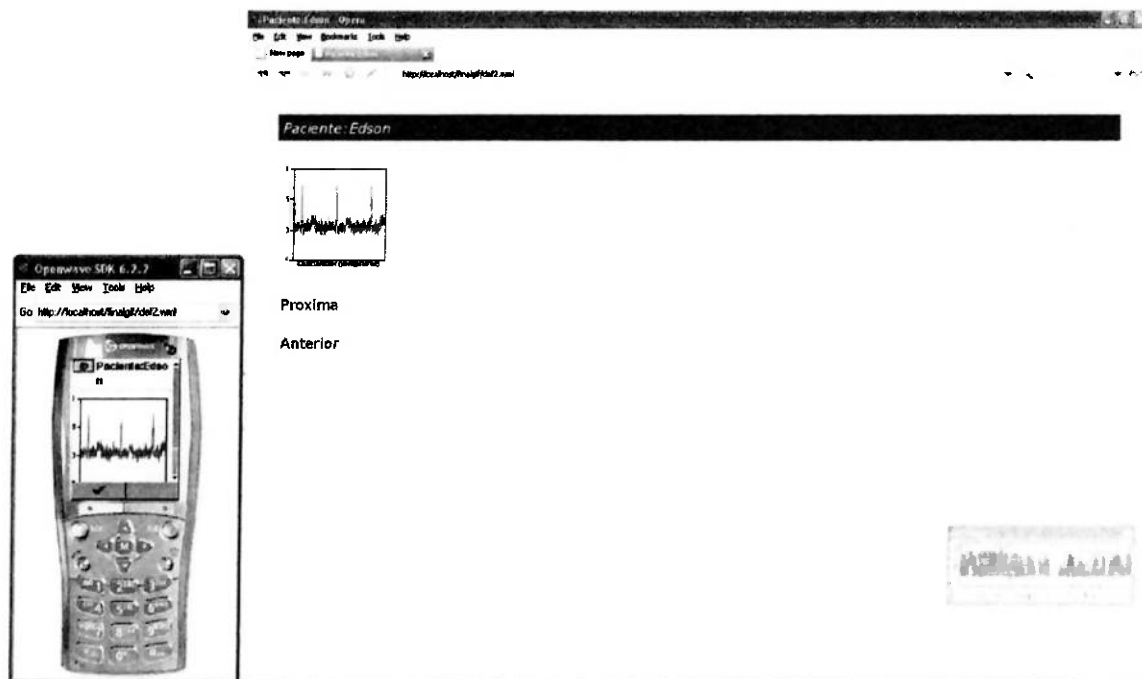


Figura 29 – Visualização da deck 2 do cenário II do paciente *Edson Mota* por meios dos dois simuladores.

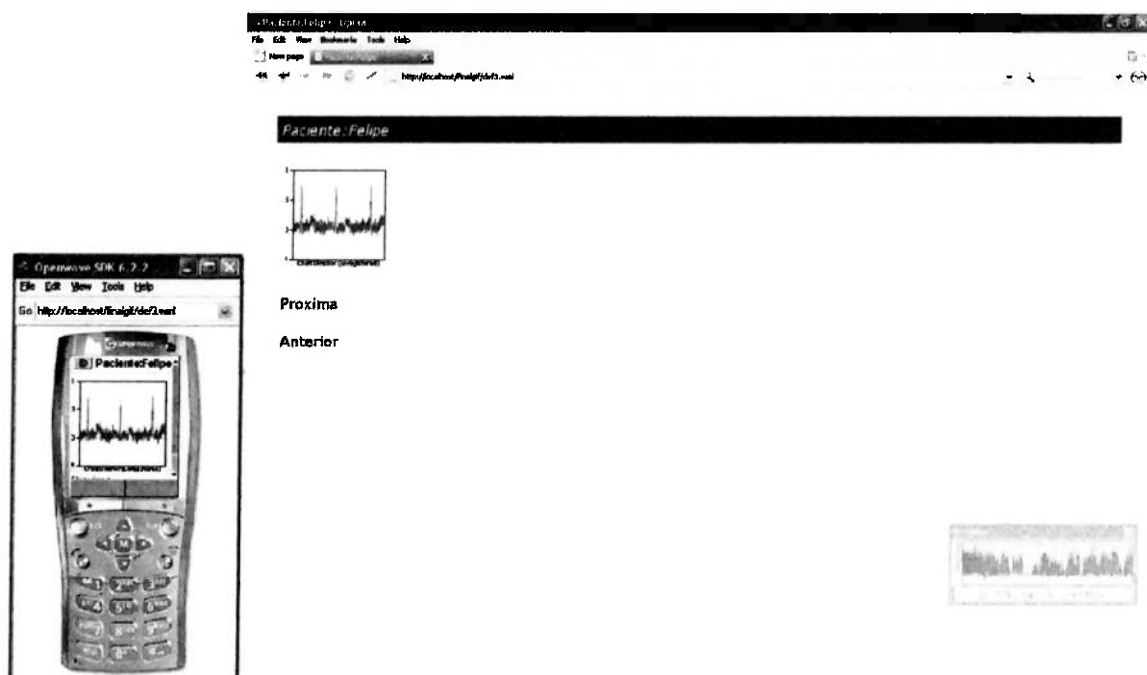


Figura 30 – Visualização da deck 2 do cenário II do paciente *Felipe Pait* por meios dos dois simuladores.

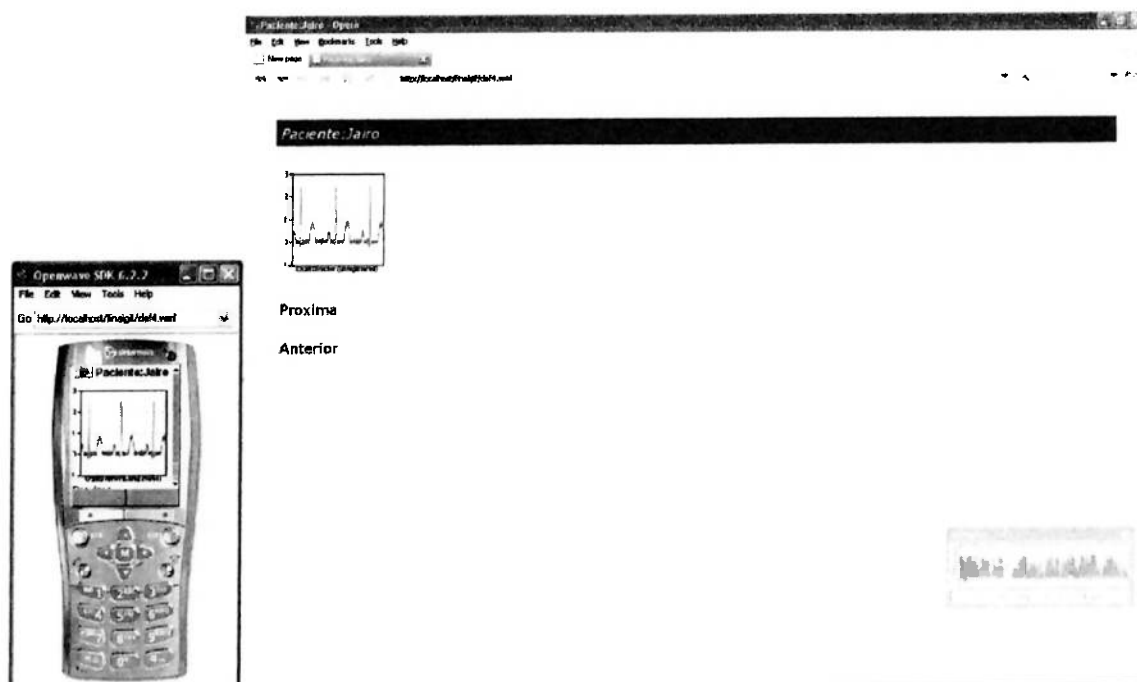


Figura 31 – Visualização da deck 2 do cenário II do paciente *Jairo Paulino* por meios dos dois simuladores.

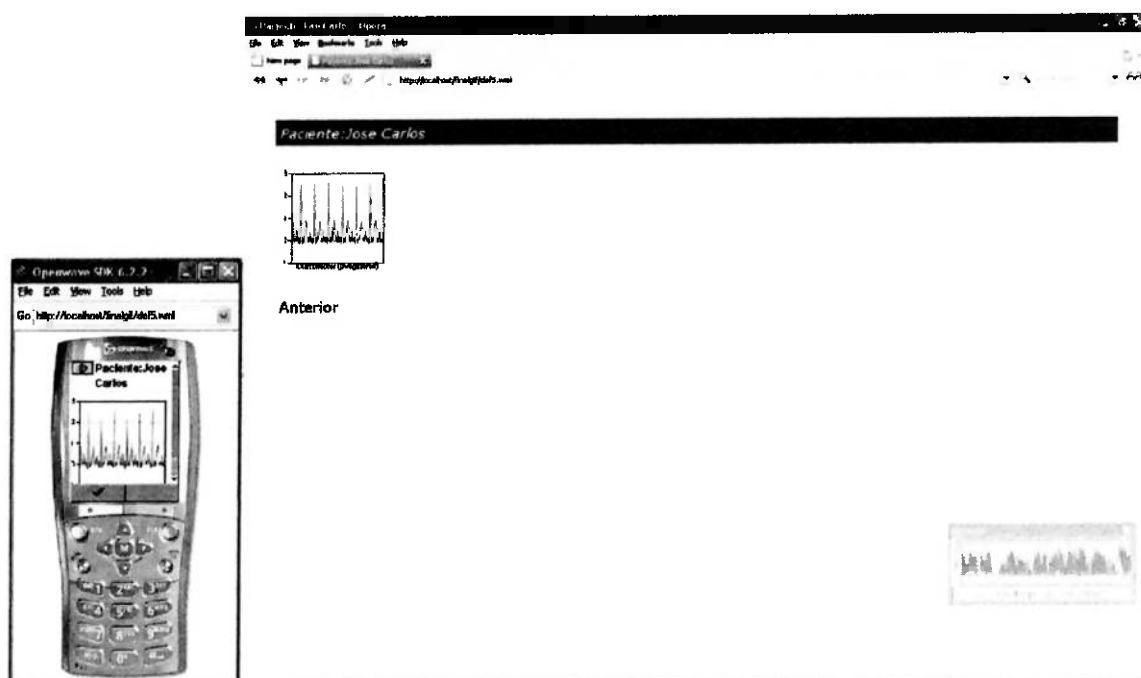


Figura 32 – Visualização da deck 2 do cenário II do paciente *José Carlos* por meios dos dois simuladores.

Foram também realizados testes utilizando os celulares Siemens C60 da operadora Tim e Motorola V600 da Operadora Claro (ambos com suporte a WAP e a imagens no formato gif). Para os 2 cenários de software do servidor, as páginas armazenadas no servidor do Leb e em um PC na residência de um dos alunos rodando Apache foram acessadas normalmente, sem ocorrer nenhum erro.

Como podemos ver, as páginas WML estão funcionando corretamente, já que os 2 simuladores abriram corretamente as páginas. Portanto, tanto o acesso do médico via WAP quanto via WEB estão funcionando conforme o esperado.

Vale chamar a atenção para o fato de que as páginas foram um pouco diferentes nos dois casos, isso devido ao jeito como o browser lê a página WML. Sendo assim, teremos páginas WML um pouco diferentes para celulares diferentes.

9. Considerações finais

Durante todo o desenvolvimento do projeto, passamos por diversas experiências que enriqueceram a nossa vasta bagagem obtida junto a Escola Politécnica da USP

Durante esse ano, aprendemos a trabalhar em equipe, administrar recursos, cumprir metas, realizar apresentações, lidar com novas tecnologias e conhecemos melhor o mundo acadêmico.

As metas estabelecidas no início do projeto foram plenamente cumpridas, apesar de pequenos ajustes que se fizeram necessários no decorrer da implementação.

A respeito da utilidade da solução obtida, acreditamos ser de grande valia em aplicações médicas, principalmente no que se refere a pacientes nos quais há uma necessidade constante de monitoramento de seu ECG devido a problemas cardíacos. A solução permite que os pacientes não tenham que se locomover exaustivas vezes ao consultório de seus médicos para os mesmos avaliarem os sinais obtidos. O médico também é beneficiado à medida que pode analisar o ECG no momento em que achar mais conveniente.

Uma sugestão para um possível aprimoramento do projeto desenvolvido seria desenvolver juntamente com o *Sistema de visualização remota de eletrocardiograma via celular*, um sistema de Análise do sinal obtido, o qual identificaria arritmias e geraria um sinal de alarme para o médico, através de, por exemplo, o envio de um SMS para o celular do médico.

Agradecimentos

Gostaríamos de agradecer a todos que de alguma maneira contribuíram para o desenvolvimento desse projeto de formatura.

Agradecemos especialmente ao nosso orientador, o Prof. José Carlos T. B. Moraes, que nos guiou durante todo esse ano. Ao LAC, principalmente ao Prof. Felipe Miguel Pait, pela seriedade e comprometimento com a coordenação das disciplinas de Laboratório de Projeto de Formatura I & II. Aos técnicos do laboratório de Engenharia Biomédica: Sandro Aureliano Miqueleti e Fernando Rangel pela grande ajuda que nos forneceram com o Hardware do Simulador de Sinais Cardíacos e com os computadores/servidores do laboratório.

Referências Bibliográficas

- [1] FAGALI G.M., MATOS H.L.V., PERON J.P.O., Relatório Final de Projeto de Formatura - Monitoração Remota de Sinais Vitais, EPUSP, 2003
- [2] FILHO J.G.G., Simulador de Sinais Cardíacos, 1999, EPUSP, 2003
- [3] FISCH C., Evalution of clinical electrocardiogram. J Am Coll Cardiol 1989
- [4] WEBSTER, J. G., Editor, Medical Instrumentation - Application and Design, 3rd Ed. John Wiley, 1998
- [5] Lynx, <http://www.lynxtec.com.br/> - Documentação da placa Lynx utilizada
- [6] Openwave Developer, http://www.openwave.com/us/products/developer_products/phone_simulator/index.htm - Simulador de celular
- [7] Apache Group, <http://www.apache.org> - Servidor Web
- [8] Wap Tiger, <http://www.waptiger.com/> - Informações sobre WAP e WML
- [9] PHP, <http://www.php.net/> - Dicas e exemplos para programadores.
- [10] Opera Browser, <http://www.opera.com> - Browser para visualização de arquivos em wml
- [11] Manuais de Cardiologia, <http://www.manuaisdecardiologia.med.br/Semiologia/ciclo1.htm> - Informações sobre o ciclo cardíaco.

Anexo A - Cronograma

Para um acompanhamento temporal das atividades desenvolvidas, os cronogramas abaixo indicam as tarefas realizadas e seus respectivos períodos ao longo do ano de 2005.

Apresentações 1 e 2

Tarefa	28/fev	7/mar	14/mar	21/mar	28/mar	4/abr	11/abr	18/abr	25/abr	2/mai	9/mai	16/mai	23/mai
Definição do Tema	x	x											
Pesquisa Bibliográfica	x	x	x	x	x	x	x	x					
Definição detalhes do projeto					x	x							
Elaboração do 1º relatório						x							
Preparação para 1ª apresentação						x							
Entrega do 1º Relatório e Apresentação							x						
Pesquisa outros projetos de Hardware								x					
Definição do Hardware									x				
Testes no Hardware										x	x	x	
Elaboração de relatório e apresentação												x	
Preparação para 2ª apresentação												x	
Entrega do 2º Relatório e Apresentação													x

Apresentações 3 e 4

Tarefa	23/mai	30/mai	6/jun	13/jun	20/jun	27/jun	1/ago	8/ago	15/ago	22/ago	29/ago	5/set	12/set	19/set
Projeto dos Softwares	x	x	x	x	x									
Elaboração do 3º relatório					x									
Preparação para 3ª apresentação					x									
Entrega do 3º Relatório e Apresentação						x								
Desenvolvimento de SW do PC							x	x	x	x	x	x	x	x
Desenvolvimento de SW do Servidor							x	x	x	x	x	x	x	x
Elaboração do 4º relatório													x	
Preparação para 4ª apresentação													x	

Apresentações 5 e 6

Tarefa	26/set	3/out	10/out	17/out	24/out	31/out	7/nov	14/nov	21/nov	28/nov	5/dez
Desenvolvimento de SW do PC	x	x	x								
Desenvolvimento de Software do Servidor	x										
Integração entre partes desenvolvidas			x	x	x	x					
Elaboração de 5º relatório				x							
Preparação para 5ª apresentação				x							
Entrega do 5º Relatório e Apresentação					x						
Testes/ Ajustes Finais						x	x	x	x	x	
Elaboração de 6º relatório										x	
Preparação para 6ª apresentação										x	
Entrega do 6º Relatório e Apresentação											x

Anexo B – Códigos-fonte Software PC

PHP - Celular

```

<?php
require_once("../lib/phpchartdir.php");

#Os dados para serem plotados no gráfico
$data = array(

#Deve-se colocar aqui os dados no formato dito anteriormente

);

#Aqui estão as amostras que se relacionam com os dados
$labels =
array(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,2
7,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52
,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,
78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,1
02,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,1
20,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,1
38,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,1
56,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,1
74,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,1
92,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,2
10,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,2
28,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,2
46,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,2
64,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,2
82,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,3
00,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,3
18,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,3
36,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,3
54,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,3
72,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,3
90,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,4
08,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,4
26,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,4
44,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,4
62,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,4
80,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,4
98,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,5
16,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,5
34,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,5
52,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,5
70,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,5

```

```

88,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,6
06,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,6
24,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,6
42,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,6
60,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,6
78,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,6
96,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,7
14,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,7
32,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,7
50,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,7
68,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,7
86,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,8
04,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,8
22,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,8
40,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,856,857,8
58,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,8
76,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,8
94,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,9
12,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,9
30,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,9
48,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,9
66,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,9
84,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999,);

```

#Cria a área total da figura com o tamanho de 150 x 150 pixels

```
$c = new XYChart(150, 150);
```

#Coloca o ponto inicial do gráfico em (10, 10) e cria a área do gráfico com o tamanho de 130 x 130 pixels

```
$c->setPlotArea(10, 10, 130, 130);
```

#Adiciona uma linha entre os dados plotados

```
$c->addLineLayer($data);
```

#Redimensiona o eixo dos dados para ocupar o gráfico de uma maneira inteligente

```
$c->xAxis->setLabels($labels);
```

#Coloca linhas de divisão no eixo dos dados

```
$c->xAxis->setLabelStep(3);
```

#Cria a imagem efetivamente no formato desejado

```
header("Content-type: image/gif");
```

```
print($c->makeChart2(GIF));
```

```
?>
```

PHP - Computador

```
<?php
require_once("../lib/phpchartdir.php");
```

```
#Os dados para serem plotados no gráfico
```

```
$data = array(
```

```
#Deve-se colocar aqui os dados no formato dito anteriormente
```

```
);
```

```
#Aqui estão as amostras que se relacionam com os dados
```

```
$labels =
```

```
array(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,2
7,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52
,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,
78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,1
02,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,1
20,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,1
38,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,1
56,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,1
74,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,1
92,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,2
10,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,2
28,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,2
46,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,2
64,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,2
82,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,3
00,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,3
18,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,3
36,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,3
54,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,3
72,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,3
90,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,4
08,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,4
26,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,4
44,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,4
62,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,4
80,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,4
98,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,5
16,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,5
34,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,5
52,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,5
70,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,5
88,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,6
06,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,6
24,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,6
42,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,6
60,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,6
```

```

78,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,6
96,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,7
14,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,7
32,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,7
50,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,7
68,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,7
86,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,8
04,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,8
22,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,8
40,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,856,857,8
58,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,8
76,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,8
94,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,9
12,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,9
30,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,9
48,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,9
66,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,9
84,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999,);

```

#Cria a área total da figura com o tamanho de 640 x 480 pixels

```
$c = new XYChart(640, 480);
```

#Coloca o ponto inicial do gráfico em (30, 20) e cria a área do gráfico com o tamanho de 590 x 430 pixels

```
$c->setPlotArea(30, 20, 590, 430);
```

#Adiciona uma linha entre os dados plotados

```
$c->addLineLayer($data);
```

#Redimensiona o eixo dos dados para ocupar o gráfico de uma maneira inteligente

```
$c->xAxis->setLabels($labels);
```

#Coloca linhas de divisão no eixo dos dados

```
$c->xAxis->setLabelStep(3);
```

#Cria a imagem efetivamente no formato desejado

```
header("Content-type: image/gif");
```

```
print($c->makeChart2(GIF));
```

```
?>
```

Captura Final

```

/*****
* Arquivo: capturafinal.c
* Descricao: Captura e processamento dos dados adquiridos
*           através da placa Lynx, assim como a criação
*           dos arquivos php que geram as imagens em GIF
* Autor: Edson Carvalho Mota Júnior
* Data: 18/08/2005 (atualização final: 28/11/2005)
*****/

#include <stdio.h>           // funcoes de entrada e saida
#include <conio.h>           // usado apenas para a funcao "getch()"
#include "placa.c"          // inclui a biblioteca com as rotinas de controle da
placa

#define NUMERO_DE_AMOSTRAS 1000 // numero de amostras

FILE *file;

main()
{
    FILE *arq_c0,*arq_c1,*arq_c2,*arq_c3,*arq_c4,*arq_c5; //arquivos php

    //declaração das variáveis

    float valor[16];

    float la[NUMERO_DE_AMOSTRAS],ra[NUMERO_DE_AMOSTRAS];
    float ll[NUMERO_DE_AMOSTRAS],rl[NUMERO_DE_AMOSTRAS];
    float lara[NUMERO_DE_AMOSTRAS],lall[NUMERO_DE_AMOSTRAS];
    float larl[NUMERO_DE_AMOSTRAS],rall[NUMERO_DE_AMOSTRAS];
    float rarl[NUMERO_DE_AMOSTRAS],llrl[NUMERO_DE_AMOSTRAS];

    int m,i;

    //escolha da frequência

    setclock(500);

    printf("Pressione qualquer tecla para iniciar a coleta de dados.\n");
    printf(" \n");

    getch();

    printf("Coletando dados...\n");

```

```
//realiza as todas as derivações

for (m=0;m<NUMERO_DE_AMOSTRAS;m++)
{
    readA2D(valor);

    la[m]=valor[0];
    ra[m]=valor[1];
    ll[m]=valor[2];
    rl[m]=valor[3];

    lara[m]=la[m]-ra[m];
    lall[m]=la[m]-ll[m];
    larl[m]=la[m]-rl[m];
    rall[m]=ra[m]-ll[m];
    rarl[m]=ra[m]-rl[m];
    llrl[m]=ll[m]-rl[m];

}

printf("Salvando arquivos...\n");

//começa a escrever os arquivos php

arq_c0=fopen("lara.php","w");
arq_c1=fopen("lall.php","w");
arq_c2=fopen("larl.php","w");
arq_c3=fopen("rall.php","w");
arq_c4=fopen("rarl.php","w");
arq_c5=fopen("llrl.php","w");

fprintf(arq_c0,"<?php\n");
fprintf(arq_c1,"<?php\n");
fprintf(arq_c2,"<?php\n");
fprintf(arq_c3,"<?php\n");
fprintf(arq_c4,"<?php\n");
fprintf(arq_c5,"<?php\n");

fprintf(arq_c0,"require_once(..lib/phpchartdir.php);\n");
fprintf(arq_c1,"require_once(..lib/phpchartdir.php);\n");
fprintf(arq_c2,"require_once(..lib/phpchartdir.php);\n");
fprintf(arq_c3,"require_once(..lib/phpchartdir.php);\n");
fprintf(arq_c4,"require_once(..lib/phpchartdir.php);\n");
fprintf(arq_c5,"require_once(..lib/phpchartdir.php);\n");
```

```
//Escreve os dados que serão desenhados
```

```
fprintf(arq_c0,"$data = array(");
fprintf(arq_c1,"$data = array(");
fprintf(arq_c2,"$data = array(");
fprintf(arq_c3,"$data = array(");
fprintf(arq_c4,"$data = array(");
fprintf(arq_c5,"$data = array(");
```

```
for (m=0;m<NUMERO_DE_AMOSTRAS;m++)
{
    fprintf(arq_c0,"%g",lara[m]);
    fprintf(arq_c1,"%g",lall[m]);
    fprintf(arq_c2,"%g",larl[m]);
    fprintf(arq_c3,"%g",rall[m]);
    fprintf(arq_c4,"%g",rarl[m]);
    fprintf(arq_c5,"%g",llrl[m]);
}
```

```
fprintf(arq_c0,");\n");
fprintf(arq_c1,");\n");
fprintf(arq_c2,");\n");
fprintf(arq_c3,");\n");
fprintf(arq_c4,");\n");
fprintf(arq_c5,");\n");
```

```
//os números das amostras que são relacionados aos dados
```

```
fprintf(arq_c0,"$labels = array\n");
fprintf(arq_c1,"$labels = array\n");
fprintf(arq_c2,"$labels = array\n");
fprintf(arq_c3,"$labels = array\n");
fprintf(arq_c4,"$labels = array\n");
fprintf(arq_c5,"$labels = array\n");
```

```
fprintf(arq_c0,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,2
```

80,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,856,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

```
fprintf(arq_c1,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,1
```

72,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,1
90,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,2
08,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,2
26,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,2
44,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,2
62,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,2
80,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,2
98,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,3
16,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,3
34,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,3
52,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,3
70,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,3
88,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,4
06,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,4
24,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,4
42,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,4
60,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,4
78,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,4
96,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,5
14,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,5
32,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,5
50,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,5
68,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,5
86,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,6
04,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,6
22,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,6
40,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,6
58,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,6
76,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,6
94,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,7
12,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,7
30,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,7
48,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,7
66,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,7
84,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,8
02,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,8
20,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,8
38,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,8
56,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,8
74,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,8
92,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,9
10,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,9
28,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,9
46,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,9
64,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,9
82,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

```
fprintf(arq_c2,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49
```

,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,
75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,1
00,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,1
18,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,1
36,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,1
54,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,1
72,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,1
90,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,2
08,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,2
26,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,2
44,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,2
62,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,2
80,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,2
98,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,3
16,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,3
34,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,3
52,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,3
70,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,3
88,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,4
06,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,4
24,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,4
42,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,4
60,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,4
78,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,4
96,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,5
14,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,5
32,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,5
50,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,5
68,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,5
86,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,6
04,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,6
22,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,6
40,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,6
58,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,6
76,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,6
94,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,7
12,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,7
30,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,7
48,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,7
66,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,7
84,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,8
02,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,8
20,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,8
38,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,8
56,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,8
74,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,8
92,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,9
10,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,9
28,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,9
46,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,9

64,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

```
fprintf(arq_c3,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,8
```

56,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

```
fprintf(arq_c4,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,7
```

48,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,7
66,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,7
84,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,8
02,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,8
20,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,8
38,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,8
56,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,8
74,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,8
92,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,9
10,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,9
28,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,9
46,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,9
64,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,9
82,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

fprintf(arq_c5,"(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,2
4,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49
,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,
75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,1
00,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,1
18,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,1
36,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,1
54,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,1
72,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,1
90,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,2
08,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,2
26,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,2
44,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,2
62,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,2
80,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,2
98,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,3
16,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,3
34,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,3
52,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,3
70,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,3
88,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,4
06,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,4
24,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,4
42,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,4
60,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,4
78,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,4
96,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,5
14,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,5
32,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,5
50,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,5
68,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,5
86,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,6
04,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,6
22,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,6

```

40,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,6
58,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,6
76,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,6
94,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,7
12,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,7
30,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,7
48,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,7
66,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,7
84,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,8
02,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,8
20,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,8
38,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,8
56,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,8
74,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,8
92,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,9
10,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,9
28,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,9
46,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,9
64,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,9
82,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999);\n");

```

```
//Cria a área da figura, com o tamanho de 640 x 480 pixels
```

```

fprintf(arq_c0,"$c = new XYChart(640, 480);\n");
fprintf(arq_c1,"$c = new XYChart(640, 480);\n");
fprintf(arq_c2,"$c = new XYChart(640, 480);\n");
fprintf(arq_c3,"$c = new XYChart(640, 480);\n");
fprintf(arq_c4,"$c = new XYChart(640, 480);\n");
fprintf(arq_c5,"$c = new XYChart(640, 480);\n");

```

```
//Coloca o gráfico para começar em (30, 20) com o tamanho de 590 x 430 pixels
```

```

fprintf(arq_c0,"$c->setPlotArea(30, 20, 590, 430);\n");
fprintf(arq_c1,"$c->setPlotArea(30, 20, 590, 430);\n");
fprintf(arq_c2,"$c->setPlotArea(30, 20, 590, 430);\n");
fprintf(arq_c3,"$c->setPlotArea(30, 20, 590, 430);\n");
fprintf(arq_c4,"$c->setPlotArea(30, 20, 590, 430);\n");
fprintf(arq_c5,"$c->setPlotArea(30, 20, 590, 430);\n");

```

```
//Adiciona uma linha entre os dados desenhados
```

```

fprintf(arq_c0,"$c->addLineLayer($data);\n");
fprintf(arq_c1,"$c->addLineLayer($data);\n");
fprintf(arq_c2,"$c->addLineLayer($data);\n");
fprintf(arq_c3,"$c->addLineLayer($data);\n");
fprintf(arq_c4,"$c->addLineLayer($data);\n");
fprintf(arq_c5,"$c->addLineLayer($data);\n");

```

```
//Realiza a autoescala do eixo x, assim como as linhas
```

```
fprintf(arq_c0,"$c->xAxis->setLabels($labels);\n");  
fprintf(arq_c1,"$c->xAxis->setLabels($labels);\n");  
fprintf(arq_c2,"$c->xAxis->setLabels($labels);\n");  
fprintf(arq_c3,"$c->xAxis->setLabels($labels);\n");  
fprintf(arq_c4,"$c->xAxis->setLabels($labels);\n");  
fprintf(arq_c5,"$c->xAxis->setLabels($labels);\n");
```

```
fprintf(arq_c0,"$c->xAxis->setLabelStep(3);\n");  
fprintf(arq_c1,"$c->xAxis->setLabelStep(3);\n");  
fprintf(arq_c2,"$c->xAxis->setLabelStep(3);\n");  
fprintf(arq_c3,"$c->xAxis->setLabelStep(3);\n");  
fprintf(arq_c4,"$c->xAxis->setLabelStep(3);\n");  
fprintf(arq_c5,"$c->xAxis->setLabelStep(3);\n");
```

```
//Cria a figura no formato GIF
```

```
fprintf(arq_c0,"header(Content-type: image/gif);\n");  
fprintf(arq_c1,"header(Content-type: image/gif);\n");  
fprintf(arq_c2,"header(Content-type: image/gif);\n");  
fprintf(arq_c3,"header(Content-type: image/gif);\n");  
fprintf(arq_c4,"header(Content-type: image/gif);\n");  
fprintf(arq_c5,"header(Content-type: image/gif);\n");
```

```
fprintf(arq_c0,"print($c->makeChart2(GIF));\n");  
fprintf(arq_c1,"print($c->makeChart2(GIF));\n");  
fprintf(arq_c2,"print($c->makeChart2(GIF));\n");  
fprintf(arq_c3,"print($c->makeChart2(GIF));\n");  
fprintf(arq_c4,"print($c->makeChart2(GIF));\n");  
fprintf(arq_c5,"print($c->makeChart2(GIF));\n");
```

```
fprintf(arq_c0,"?>");  
fprintf(arq_c1,"?>");  
fprintf(arq_c2,"?>");  
fprintf(arq_c3,"?>");  
fprintf(arq_c4,"?>");  
fprintf(arq_c5,"?>");
```

```
//fecha e finaliza os arquivos php
```

```
fclose(arq_c0);  
fclose(arq_c1);  
fclose(arq_c2);  
fclose(arq_c3);  
fclose(arq_c4);  
fclose(arq_c5);  
printf("Fim.\n");
```

```
}
```

Controle da Placa Lynx

```

/* BIBLIOTECA DE ROTINAS PARA CONTROLE DA PLACA LYNX 10/26 */
/* Esta versao: 18/08/2005 (ECMota) */

#include <stdio.h>
#include <math.h>
#include <dos.h>
#include <conio.h>
#include <process.h>

#define END_PLACA 720 // endereco da placa cad 10/26 no PC
#define PALAVRA_CONTROLE 180 // palavra de controle do temporizador
// (ver pg.1-7 do manual)
#define FREQ_MICRO 8000000 // frequencia do micro

/*void setclock(long freq);
void readA2D(float valor[16]);
void writeD2A(double Canal1,double Canal2);
*/
void setclock(long freq)
{
    unsigned int contagem;
    unsigned char meubyte;

    outp(END_PLACA+3,PALAVRA_CONTROLE);
    contagem = (unsigned int)((FREQ_MICRO/4)/freq);
    meubyte = (unsigned char) contagem;
    outp(END_PLACA+2,meubyte);
    meubyte = (unsigned char) (contagem >> 8);
    outp(END_PLACA+2,meubyte);
}

void readA2D(float valor[16])
{
    int i,aux[16];
    unsigned char compara=16;

    while((inp(END_PLACA+5) & compara) == 0);
    while((inp(END_PLACA+5) & compara) !=0);

    for(i=0;i<16;i++)
    {
        outp(END_PLACA + 7,240+i);
        while((inp(END_PLACA+5) & 16)!=0);
        aux[i] = ((inp(END_PLACA+5) & 3)*256 + inp(END_PLACA + 4));
        valor[i] = (float) 10*(aux[i] - 512) / 1024;
    }
}

```

```
void writeD2A(float Canal1,float Canal2)
{
    int aux1,aux2,end;
    unsigned char val_placa;

    /* Saturacao da saida */
    if ((Canal1 >= -5) && (Canal1 <= 5) && (Canal2 >= -5)
        && (Canal2 <= 5))
    {
        aux1 = fabs(((Canal1 + 5)*256)/10);
        aux2 = fabs(((Canal2 + 5)*256)/10);

        /* END_PLACA+5 : Canal B do conversor D/A */
        end = END_PLACA + 5;
        val_placa = aux1;
        outp(end,val_placa);

        /* END_PLACA+4 : Canal A do conversor D/A */
        end = END_PLACA + 4;
        val_placa = aux2;
        outp(end,val_placa);
    }
}
```

Anexo C – Códigos-fonte Software Servidor

CENÁRIO I

Deck 1 – Apresentação

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<card id="lista" title="Pacientes">
<p>Escolha o Paciente:<br />
<br />
<a href="edson.wml"> Edson Mota </a> <br />
<a href="felipe.wml"> Felipe Pait </a> <br />
<a href="jairo.wml"> Jairo Paulino </a> <br />
<a href="josecarlos.wml"> Jose Carlos </a> <br />
</p>

<do type="accept" label="Instrucoes">
<go href="#Instrucoes"/>
</do>

</card>

<card id="Instrucoes">
<p>Bemvindo ao Visualizador Remoto</p>
<p>de eletrocardiograma via celular</p>
<p>Escolha o nome do paciente desejado</p>

</card>

</wml>
```

Deck 2 – Figuras ECG

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>
<card id="card2" title="edson">

<p>
Paciente:Edson Mota
</p>

<p>
```

```

</p>
```

```
<p>
  <anchor>
    Anterior
  </prev/>
</anchor>
</p>
```

```
</card>
```

```
</wml>
```

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
<card id="card3" title="felipe">
```

```
<p>
Paciente:Felipe Pait
```

```
</p>
```

```
<p>
```

```

```

```
</p>
```

```
<p>
  <anchor>
    Anterior
  </prev/>
</anchor>
</p>
```

```
</card>
```

```
</wml>
```

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
<card id="card4" title="Paciente:Jairo">
```

```
<p>
Paciente:Jairo Paulino
</p>
```

```
<p>

</p>
```

```
<p>
<anchor>
  Anterior
<prev/>
</anchor>
</p>
</card>
</wml>
```

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
<card id="card4" title="josecarlos">
```

```
<p>
Paciente:Jose Carlos
</p>
```

```
<p>

</p>
```

```
<p>
<anchor>
  Anterior
<prev/>
</anchor>
</p>
</card>
</wml>
```

CENÁRIO II

Deck 1 – Apresentação

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
```

```
<card title="Apresentacao">
```

```
<p>
<anchor>Mostrar Figuras
<go href="def2.wml"/>
</anchor>
</p>
```

```
<do type="accept" label="Instrucoes">
<go href="#Instrucoes"/>
</do>
```

```
</card>
```

```
<card id="Instrucoes">
<p>Bemvindo ao Visualizador Remoto</p>
<p>de ecg via celular</p>
<p>Percorra pelas imagens</p>
<p>dentre as opcoes fornecidas</p>
```

```
</card>
```

```
</wml>
```

Deck 2 – Figuras ECG

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
```

```
<card id="card2" title="Paciente:Edson">
```

```
<p>
```

```

```

```
</p>
```

```

<p>
<anchor>Proxima
<go href="def3.wml"/>
</anchor>
</p>
<p>
<anchor>
Anterior
<prev/>
</anchor>
</p>

</card>

</wml>

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>
<card id="card3" title="Paciente:Felipe">

<p>



</p>

<p>
<anchor>Proxima
<go href="def4.wml"/>
</anchor>
</p>
<p>
<anchor>
Anterior
<prev/>
</anchor>
</p>

</card>
</wml>

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>
<card id="card4" title="Paciente:Jairo">

```

<p>

</p>

<p>

<anchor>Proxima

<go href="def5.wml"/>

</anchor>

</p>

<p>

<anchor>

Anterior

<prev/>

</anchor>

</p>

</card>

</wml>

<?xml version="1.0"?>

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"

"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>

<card id="card5" title="Paciente:Jose Carlos">

<p>

</p>

<p>

<anchor>

Anterior

<prev/>

</anchor>

</p>

</card>

</wml>