



**UNIVERSIDADE LUTERANA DO BRASIL
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**



FÁBIO DE ARAÚJO BAIROS

**UTILIZAÇÃO DE SINAIS MIOELÉTRICOS
NO COMANDO DE PRÓTESES BIOMECÂNICAS**

**CANOAS
2008**

FÁBIO DE ARAÚJO BAIROS

**UTILIZAÇÃO DE SINAIS MIOELÉTRICOS
NO COMANDO DE PRÓTESES BIOMECÂNICAS**

Trabalho de Conclusão de Curso apresentado ao Departamento de Engenharia Elétrica da ULBRA como um dos requisitos obrigatórios para a obtenção do grau de Engenheiro Eletricista

Departamento: Engenharia Elétrica

Área de Concentração: Biomédicas.

Professor Orientador: Eng. Eletr. Paulo César Cardoso Godoy – CREA-RS: 011.6822-D

**CANOAS
2008**

FOLHA DE APROVAÇÃO

Nome do Autor: Fabio de Araújo Bairros

Matrícula: 011100913-8

Título: UTILIZAÇÃO DE SINAIS MIOELÉTRICOS NO COMANDO DE PRÓTESES BIOMECÂNICAS.

Trabalho de Conclusão de Curso apresentado ao Departamento de Engenharia Elétrica da ULBRA como um dos requisitos obrigatórios para a obtenção do grau de Engenheiro Eletricista

Professor Orientador:

Eng. Eletr. Paulo César Cardoso Godoy – CREA-RS: 011.6822-D

Banca Avaliadora:

Eng. Eletr. Marília Amaral da Silveira

CREA-RS: RS050909-D

Eng. Eletr. Valner João Brusamarello

CREA-RS: RS078158-D

Assinaturas:

Autor

Fábio de Araújo Bairros

Orientador

Paulo César Cardoso Godoy

Avaliador

Marília Amaral da Silveira

Avaliador

Valner João Brusamarello

Relatório Aprovado em:

Dedico a todos que me apoiaram nesta empreitada.

AGRADECIMENTOS

A todos que colaboraram na elaboração deste trabalho, o meu reconhecimento.

Aos Professores Alexandre Balbinot e Paulo César Godoy pelo estímulo, dedicação e esforço pessoal proporcionado.

A todos os colegas e professores pelas sugestões e observações valiosas.

*"Não basta saber, é preciso também aplicar;
não basta querer, é preciso também agir."
Goethe*

RESUMO

DE ARAÚJO BAIROS, Fabio. **Utilização de sinais miolétricos no comando de próteses biomecânicas: Bioprótese.** Trabalho de Conclusão de Curso em Engenharia Elétrica - Departamento de Engenharia Elétrica. Universidade Luterana do Brasil. Canoas, RS. 2007.

A comunidade científica, principalmente ligada a área de biomecânica vem, ao longo das últimas décadas, realizando pesquisas na elaboração de próteses bioativas comandadas pelos sinais bioelétricos humanos. O objetivo é atender indivíduos que possuam necessidades especiais de locomoção, proporcionando lhes um aumento no seu grau de independência.

Os objetivos deste trabalho são implementar um dispositivo que permita ler sinais bioelétricos, proveniente dos músculos do antebraço humano, realizar a sua interpretação e comandar uma estrutura mecânica que simule uma prótese da mão humana. Para este fim, realizou-se a leitura dos sinais mioelétricos com o uso de eletrodos não-invasivos, efetuado sua digitalização. Com o uso de ferramentas matemáticas e técnicas já consagradas como FFT, foi realizada a interpretação das informações e como atuação final do processo realizou o comando da estrutura mecânica, também desenvolvida ao longo do trabalho.

Com os testes realizados foi possível avaliar o movimento dos músculos de flexão profunda dos dedos e de flexão ulnar do carpo. Através dos sinais obtidos destes movimentos foi possível atuar parcialmente na prótese desenvolvida.

Palavras-chave: biomecânica, bioativas, bioelétricos, prótese, mioelétricos.

ABSTRACT

DE ARAÚJO BAIROS, Fábio. **The use of myoelectric signals on the command of biomechanical prosthesis: Bioprosthesis.** Electrical Engineering Monography – Department of Electrical Engineering. Universidade Luterana do Brasil. Canoas, RS.2008

The scientific community, related to the biomechanical field, has been conducting research on the elaboration of bioactive prosthesis commanded by human bioelectrical signals. It aims at helping individuals, who have special locomotion needs, providing them with higher independence.

The objective of this work consists of the implementation of a device which allows the reading of bioelectrical signals proceeding from the human forearm, interpreting and commanding a mechanical structure which simulates a human hand prosthesis. The myoelectric signals were collected with noninvasive electrodes. After digitalizing the signal was processed with mathematical tools such a FFT. After that, all the information was interpreted and the myoelectric signal commands a mechanical structure, developed along the project as well.

With the tests were possible to evaluate the movement of the fingers deep flexion muscle and wrist deep flexion muscle. With this signals were possible to act in part of the prosthesis.

Key words: biomechanics, bioactive, bioelectrical, prosthesis, myoelectric.

LISTA DE FIGURAS

Figura 1	Estrutura muscular do antebraço humano	18
Figura 2	Aspectos mecânicos do movimento muscular, actina e miosina atuando	19
Figura 3	Movimento de adução da mão humana	20
Figura 4	Movimento de flexão dos dedos	20
Figura 5	Músculo flexor ulnar do carpo	21
Figura 6	Músculo flexor profundo dos dedos	21
Figura 7	Sinal EMG no domínio da frequência	22
Figura 8	Potencial de ação típico	23
Figura 9	Imagem do ventre muscular com eletrodo posicionado	24
Figura 10	Posicionamento dos eletrodos para aquisição do sinal mioelétrico	25
Figura 11	Eletrodos não invasivos reutilizáveis	25
Figura 12	Exemplo de digitalização de um sinal analógico qualquer	26
Figura 13	Organização da FFT	28
Figura 14	Exemplo de janela de Hamming	29
Figura 15.A	Músculo flexor profundo dos dedos	30
Figura 15.B	Movimento de flexão dos dedos	30
Figura 16.A	Músculo flexor ulnar do carpo	30
Figura 16.B	Movimento de adução da mão	30
Figura 17	Diagrama em blocos do projeto	31
Figura 18	Um par de eletrodos	32
Figura 19	Eletrodos posicionados sobre a pele	32
Figura 20.A	Eletrodos posicionados sobre o músculo flexor profundo dos dedos	33
Figura 20.B	Eletrodos sobre o músculo flexor ulnar do carpo	33
Figura 21	Diagrama do condicionador para um canal	34
Figura 22	Interface de aquisição para um canal	35
Figura 23	Fluxograma do programa rodando no PIC16F72	35
Figura 24	Diagrama em blocos da unidade central	36
Figura 25	Fluxograma da unidade central	37
Figura 26	Tela principal do programa	38
Figura 27	Diagrama etapa de potência	39
Figura 28	Fluxograma etapa de potência	40
Figura 29	Projeto gráfico da mão mecânica	41
Figura 30	Mão mecânica montada	42
Figura 31	Anteparo preso na haste interrompendo o sinal do sensor fotoelétrico	42
Figura 32.A	Colete vista traseira	43
Figura 32.B	Colete vista lateral sendo vestido pelo usuário	43
Figura 33	Diagrama de teste da etapa de condicionamento	44
Figura 34.A	Teste do conversor AD sentido crescente	46
Figura 34.B	Teste do conversor AD sentido decrescente	46
Figura 35	Teste do aplicativo para sinal de 10 Hz	46
Figura 36	Teste do aplicativo para sinal de 50 Hz	47
Figura 37	Teste do aplicativo para sinal de 90 Hz	47
Figura 38	Imagem do ensaio no músculo flexor profundo dos dedos	49
Figura 39	Gráfico dos ensaios no músculo flexor profundo dos dedos no indivíduo #1	50
Figura 40	Imagem do ensaio no músculo flexor ulnar do carpo	50
Figura 41	Gráfico dos ensaios no músculo flexor ulnar indivíduo #1	51

Figura 42	Gráfico dos ensaios no músculo flexor profundo dos dedos no indivíduo #2..	52
Figura 43	Gráfico dos ensaios no músculo flexor ulnar indivíduo #2	53
Figura 44	Gráfico dos ensaios no músculo flexor profundo dos dedos no indivíduo #3..	54
Figura 45	Gráfico dos ensaios no músculo flexor ulnar indivíduo #3	55
Figura 46.A	Haste na posição inicial	58
Figura 46.B	Haste na posição final.....	58

LISTA DE TABELAS

Tabela 1	Ensaio na conversão sentido crescente	45
Tabela 2	Ensaio na conversão sentido decrescente	45
Tabela 3	Ensaio de medição da flexão profunda dos dedos no indivíduo #1.....	49
Tabela 4	Ensaio de medição da flexão ulnar do carpo no indivíduo #1.....	51
Tabela 5	Ensaio de medição da flexão profunda dos dedos no indivíduo #2.....	52
Tabela 6	Ensaio de medição da flexão ulnar no indivíduo #2.....	53
Tabela 7	Ensaio de medição da flexão profunda dos dedos no indivíduo #3.....	54
Tabela 8	Ensaio de medição da flexão ulnar do carpo no indivíduo #3.....	55
Tabela 9	Desvio padrão para os sinais do músculo flexor profundo dos dedos.....	56
Tabela 10	Desvio padrão para os sinais do músculo flexor ulnar do carpo	56
Tabela 11	Resultados do teste de funcionamento do protótipo para um tempo de 10 segundos	58
Tabela 12	Resultados do teste de funcionamento do protótipo para um tempo de 5 segundos	59

LISTA DE ABREVIATURAS E SIGLAS

- ABNT: Associação Brasileira de Normas Técnicas.
- ADC: Analog to Digital Converter
- DFT: Discrete Fourier Transform
- FFT: Fast Fourier Transform
- NBR: Norma Brasileira de Regulamentação
- SNC: Sistema Nervoso Central

LISTA DE SÍMBOLOS

Ah: Ampére-hora

Ca: elemento químico cálcio

Hz: Hertz

Na: elemento químico sódio

K: elemento químico potássio

V: Volt

SUMÁRIO

1 INTRODUÇÃO	15
1.1 JUSTIFICATIVA	16
1.2 OBJETIVOS GERAIS.....	16
1.3 OBJETIVOS ESPECÍFICOS	16
1.4 METAS	16
2 REVISÃO BIBLIOGRÁFICA	17
2.1 FISIOLOGIA DO MÚSCULO ESQUELÉTICO ESTRIADO HUMANO	17
2.2 CINESIOLOGIA	19
2.3 ELETROMIOGRAFIA	22
2.3.1 Potencial de ação muscular.....	22
2.3.2 Coleta do sinal mioelétrico.....	24
2.3.3 Eletrodos.....	25
2.4 PROCESSAMENTO DIGITAL DO SINAL MIOLÉTRICO	26
3 DESENVOLVIMENTO DA BIOPRÓTESE.....	30
3.1 ELETRODOS E POSICIONAMENTO	32
3.2 CONDICIONAMENTO E INTERFACE DE AQUISIÇÃO.....	33
3.3 UNIDADE CENTRAL.....	36
3.4 INTERFACE DE POTÊNCIA E MÃO MECÂNICA.....	39
3.5 DISPOSITIVO AUXILIAR.....	42
4 RESULTADOS E DISCUSSÕES	44
4.1 TESTES FUNCIONAIS DAS PARTES.....	44
4.2 TESTES DOS SINAIS MIOLÉTRICOS	48
4.3 TESTES NO SISTEMA INTERLIGADO.....	57
5 CONCLUSÃO.....	60
REFERÊNCIAS	62
OBRAS CONSULTADAS.....	64
APÊNDICES.....	65
Apêndice A – Fonte programa da interface de potência.....	66
Apêndice B – Fonte programa da interface de aquisição	74
Apêndice C – Fonte programa da unidade central	76
Apêndice D: Esquema elétrico – Interface de aquisição – Digital.....	105
Apêndice E: Esquema elétrico – Interface de aquisição – Analógica	106
Apêndice F: Esquema elétrico – Interface de potência – Drivers	107
Apêndice G: Fluxograma de testes na interface de potência e mão mecânica	108
Apêndice H: Fonte do programa de testes da interface de potência.....	109
Apêndice I: Cronograma de atividades	116
ANEXOS	117
Anexo A: Data-sheet – Amplificador de instrumentação - INA 101 – Texas Instruments – Burr-Brown.....	118

Anexo B: Data-sheet – Microcontrolador – PIC 16F72 – Microchip	119
Anexo C: Data-sheet – Interface RS-232 – HIN 232 – Intersil.....	120
Anexo D: Data-sheet – Amplificador operacional – OPA 4227 – Texas Instruments – Burr Brown.....	122
Anexo E: Data-sheet – Motor de passo – PM 35 L– Minibea.....	123
Anexo F: Data-sheet – Motor de passo – PM 55 L– Minibea.....	124

1 INTRODUÇÃO

Este trabalho de conclusão descreve as atividades de pesquisa e implementação de um dispositivo capaz de efetuar a leitura dos sinais elétricos musculares (mioelétricos). Também atua na interpretação destes sinais e na sua repetição para o acionamento de um mecanismo que emula a atividade da mão humana. A principal motivação, para o início desta atividade, está na possibilidade de atender indivíduos com necessidades especiais.

O capítulo dois faz uma síntese do conhecimento produzido no âmbito da biomecânica, eletromiografia, de pesquisas que colaboraram em descobertas científicas. Também é feita uma descrição resumida dos conceitos e técnicas empregados no desenvolvimento deste projeto e dos tipos de movimentos analisados.

No capítulo três é feita a descrição, em uma abordagem estruturada a projetos, de todas as etapas que envolveram a implementação. Estas etapas estão divididas em pesquisa, elaboração, projeto e construção. A etapa de pesquisa caracteriza-se pela fase de coleta de informações das mais diversas fontes, desde a literatura clássica até as visitas e entrevistas com profissionais das áreas de engenharia e medicina. Na fase de elaboração tem-se uma descrição global das metas a serem atingidas durante o decorrer do trabalho e a definição de uma estrutura básica em forma de diagrama de blocos. Na etapa de projeto é feita a descrição de cada componente envolvido na implementação dos blocos e a definição dos requisitos de cada item. Finalizando, a etapa de construção descreve as atividades de montagem dos componentes individuais, sua integração e atividades de testes.

No capítulo quatro tem-se uma relação de informações coletadas nas etapas de testes e validação da estrutura montada. Nesta seção é feito o levantamento das informações com o auxílio de ferramentas de estatística e análise.

No capítulo cinco é feita uma abordagem dos resultados obtidos, levantamento do que foi absorvido de conhecimento em cada etapa, pontos a serem melhorados tanto em termos de custo como também em termos funcionais. Neste ponto também se destaca a real importância do que foi desenvolvido até o momento, ressaltando as aplicações, tanto no meio acadêmico, como no cotidiano da sociedade.

1.1 JUSTIFICATIVA

O projeto tem como principal justificativa suprir a carência do mercado brasileiro no que se refere a próteses comandadas a partir dos sinais mioelétricos. Contribuir na pesquisa de soluções alternativas de baixo custo, pouca manutenção e fácil reposição também são justificativas para o desenvolvimento do protótipo.

1.2 OBJETIVOS GERAIS

O projeto tem como objetivo geral desenvolver uma prótese que possa ser comandada a partir da interpretação dos sinais mioelétricos humanos.

1.3 OBJETIVOS ESPECÍFICOS

Desenvolver um protótipo que possa efetuar a aquisição dos sinais mioelétricos de 2 músculos esqueléticos estriados, realizar o processamento destas informações de maneira a identificar os movimentos musculares e comandar uma prótese mecânica com 2 motores representando os movimentos.

1.4 METAS

Construir um protótipo que possa efetuar a leitura dos sinais mioelétricos dos músculos flexor profundo dos dedos e flexor ulnar do carpo. Aplicar as ferramentas de processamento matemático FFT e janelamento de Hamming a fim de interpretar estes sinais. Comandar 2 motores de passo pertencentes a um dispositivo mecânico que compõem uma prótese mecânica.

2 REVISÃO BIBLIOGRÁFICA

2.1 FISIOLOGIA DO MÚSCULO ESQUELÉTICO ESTRIADO HUMANO

A fisiologia é um segmento da biologia que se preocupa em descrever o funcionamento das partes, sistemas e ou conjuntos de elementos afins, que formam um determinado organismo vivo. Esta descrição leva em consideração aspectos físicos e químicos responsáveis pela origem e manutenção da vida (GUYTON, 1992).

Os músculos esqueléticos estriados são responsáveis pelo movimento voluntário de ossos do sistema esquelético. Este tipo de movimento é aquele executado quando o indivíduo se encontra em estado consciente. Os músculos esqueléticos estão associados fisicamente aos ossos, de maneira que o movimento é interdependente já que o sistema esquelético dá sustentação a todo o conjunto proporcionando maior resistência mecânica. Este tipo de músculo ocupa cerca de 40 %, na média, da massa corporal do ser humano e tem coloração avermelhada devido as suas componentes (GUYTON, 1992).

O músculo esquelético é do tipo estriado porque seu formato é caracterizado por um aglomerado de inúmeros micro-tubos alongados denominados de miofibrilas. Estes possuem propriedades elásticas em sua constituição o que promove, no conjunto, o movimento de toda a formação (GUYTON, 1992). Fazendo uma analogia, as miofibrilas têm sua aparência semelhante a um cabo de um condutor de eletricidade que é constituído por inúmeros fios. Com relação a sua funcionalidade, pode-se traçar um paralelo com materiais sintéticos industrializados, como algumas borrachas que tem a propriedade de poderem ser deformadas.

A figura 1 apresenta uma vista expandida da estrutura muscular desde sua formação molecular até a inserção em um determinado osso. Neste caso específico o osso em questão é a falange média do dedo indicador da mão direita. O músculo representado é o flexor profundo dos dedos e nesta figura se observa duas componentes básicas que compõem as miofibrilas, que são a miosina e a actina.

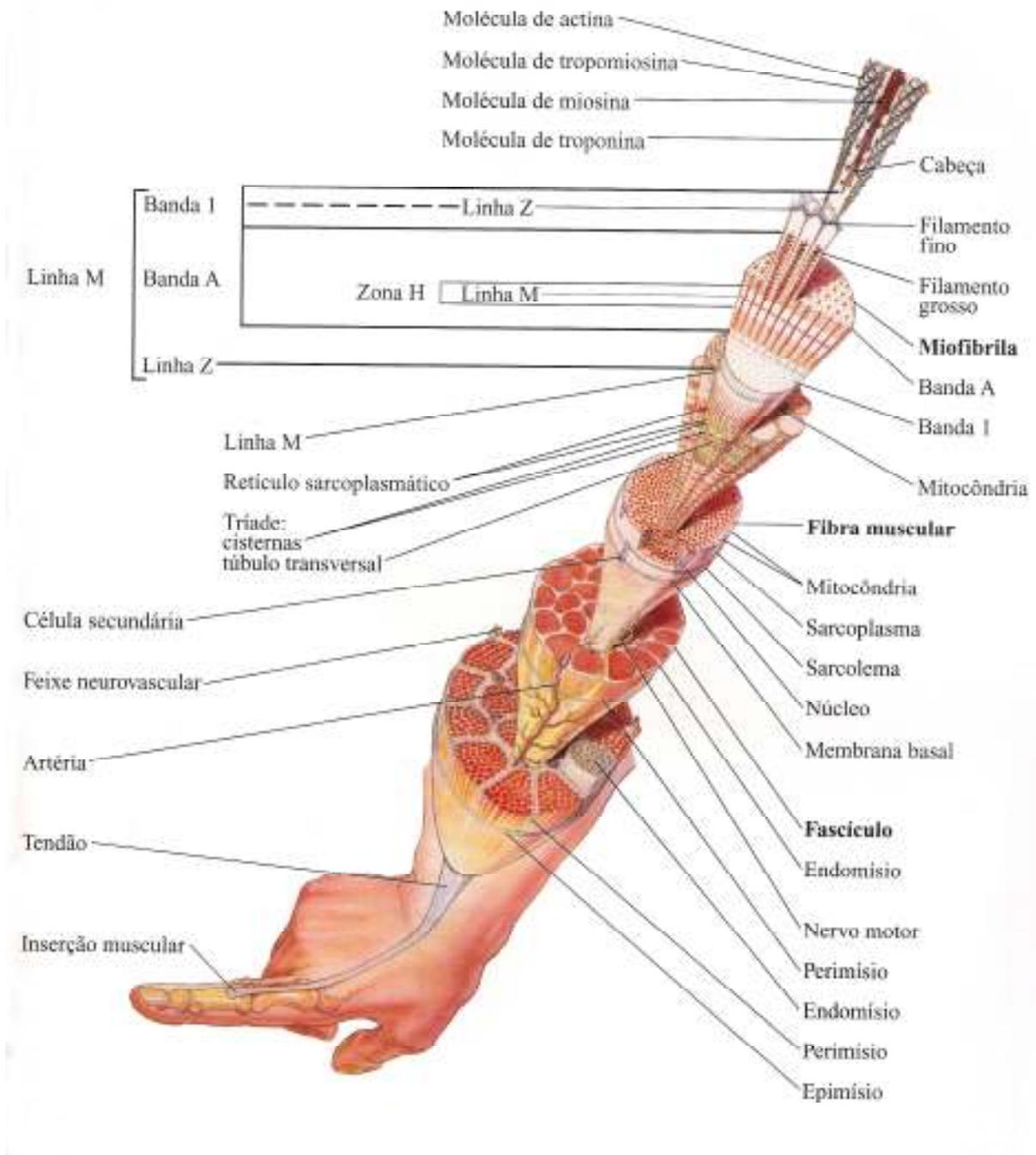


Figura 1: Estrutura muscular do antebraço humano (PEREZ, 2005).

O que determina o encurtamento ou contração e o relaxamento ou tração das miofibrilas e por conseqüência, do músculo, é o movimento relativo entre a miosina e a actina já que existem pequenos espaçamentos onde é possível acomodar estas estruturas quando o músculo está executando algum tipo de movimento (GUYTON, 1992).

Na figura 2 é possível verificar as duas componentes da miofibrila atuando quando o músculo está relaxado ou quando o mesmo se encontra contraído.

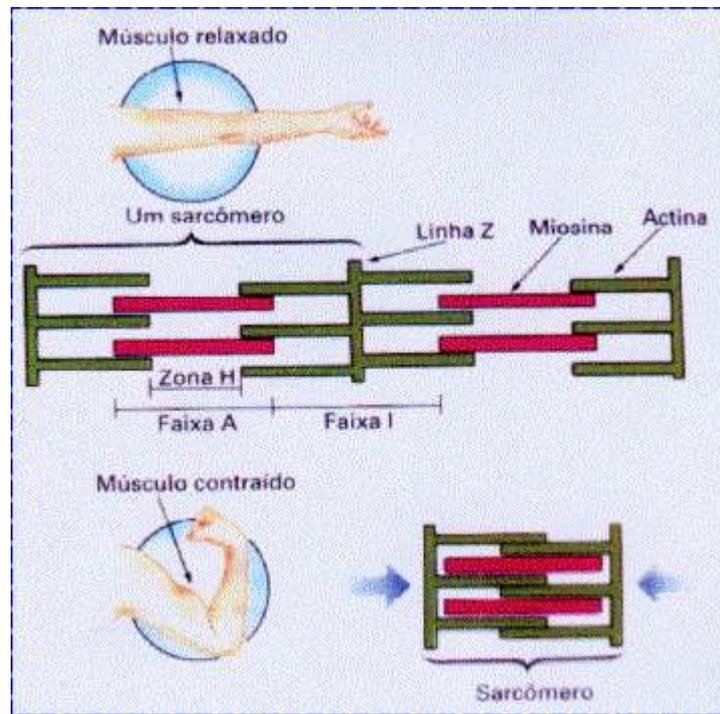


Figura 2: Aspectos mecânicos do movimento muscular, actina e miosina atuando (VILELA, 2008).

O movimento relativo entre a miosina e a actina é provocado por um sinal denominado de potencial de ação, sinal elétrico, proveniente do moto-neurônio. O potencial de ação percorre as fibras musculares, pois estas se encontram imersas em substâncias neurotransmissoras a base de íons de cálcio (Ca), sódio (Na) e potássio (K). Este potencial de ação, em última análise, gera forças atrativas entre os filamentos da actina e miosina, durante o processo de contração muscular. Esta contração se repete em cada um dos inúmeros filamentos acarretando um encurtamento geral do músculo (GUYTON, 1992).

2.2 CINESIOLOGIA

A cinesiologia estuda os movimentos dos seres vivos sobre seu aspecto visual efetuando definições, padronizações nas nomenclaturas e estuda a mecânica das forças envolvidas nestes movimentos (MIRANDA, 2006).

Praticamente todos os músculos que proporcionam o movimento da mão humana estão alojados no antebraço (DÂNGELO, 2002). Os movimentos que serão estudados são os de adução da mão e o de flexão das falanges (dedos).

A figura 3 indica o movimento radial da mão humana, também denominado de adução. Na figura verifica-se a mão esquerda realizando o movimento e o músculo responsável é o flexor ulnar do carpo.



Figura 3: Movimento de adução da mão humana (DÂNGELO, 2002).

A figura 4 indica o movimento de flexão dos dedos, exceto do polegar. Neste tipo de movimento o músculo responsável é o flexor profundo dos dedos.

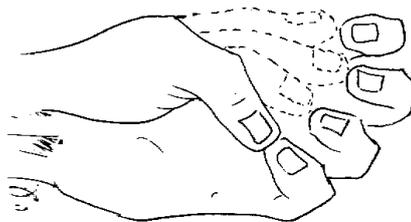


Figura 4: Movimento de flexão dos dedos (DÂNGELO, 2002).

Os músculos pesquisados são responsáveis pelo fechamento da mão (flexão dos dedos, exceto polegar) e pelo movimento de adução da mão (movimento axial). Estes músculos estão localizados no antebraço.

A figura 5 apresenta um dos músculos estudados no projeto e sua localização no antebraço humano. Trata-se do músculo flexor radial (ulnar). A figura 6 mostra o músculo flexor profundo dos dedos.

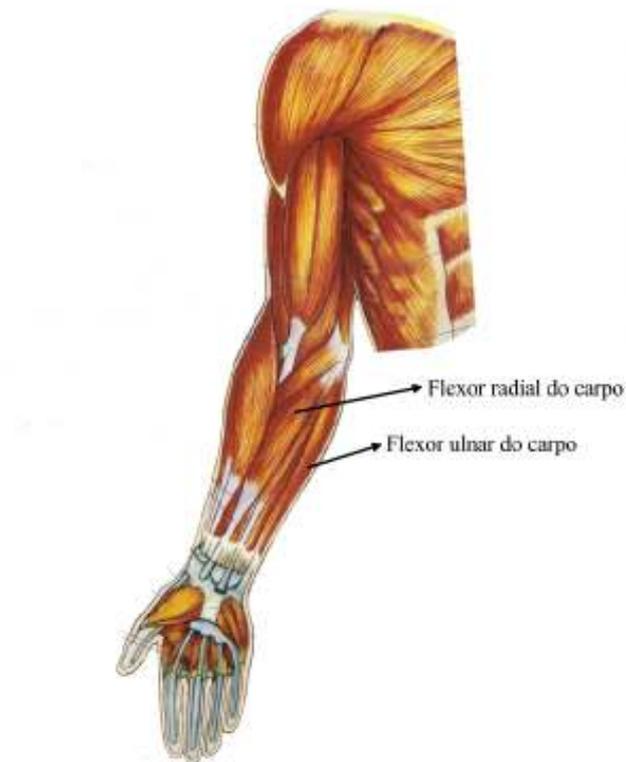


Figura 5: Músculo flexor ulnar do carpo (PEREZ, 2005).

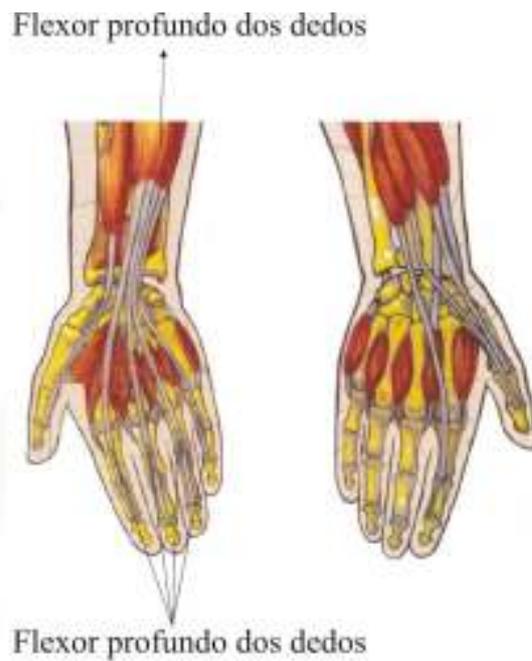


Figura 6: Músculo flexor profundo dos dedos (PEREZ, 2005).

2.3 ELETROMIOGRAFIA

A eletromiografia, ou EMG, se preocupa com a observação, estudo e registro das atividades bioelétricas que ocorrem nas fibras musculares esqueléticas estriadas (DE LUCA, 2002).

A figura 7 representa um sinal mioelétrico da atividade do músculo esquelético. Este sinal tem predominância de intensidade no espectro de frequências abaixo de 100 Hz.

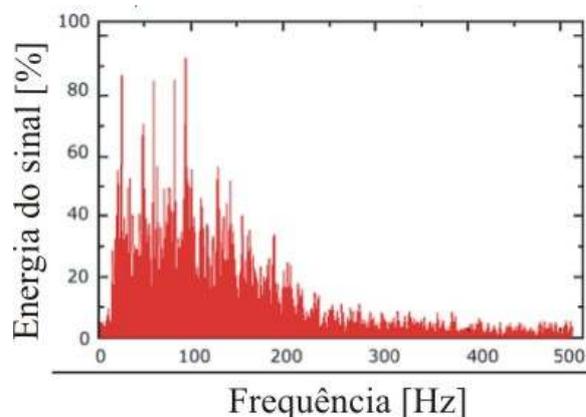


Figura 7: Sinal EMG no domínio da frequência (LUCA, 2002).

Todo o processo para acionamento do músculo esquelético parte do sistema nervoso central (SNC), do córtex motor, percorre a medula espinhal e os ramos nervosos dos membros superiores (GUYTON, 1992).

O impulso nervoso chega aos músculos através do moto-neurônio, que juntamente com determinados nervos formam a placa motora, que é onde ocorre a sinapse para o movimento. Sinapse é nome dado às interligações neuronais com o objetivo da transmissão das informações que, fisicamente, são os impulsos bioelétricos (GUYTON, 1993).

2.3.1 Potencial de ação muscular

O potencial de ação muscular, ou seja, aquele que chega ao elemento final (fibras musculares) possui um comportamento semelhante ao sinal da origem, no córtex motor. A diferença entre estes é a intensidade do mesmo. O sinal que vem do SNC é de pequena

intensidade, cerca de dezenas de micro-volts, enquanto o que chega às fibras musculares é cerca de mil vezes mais, em torno de dezenas de mili-volts (GUYTON, 1993).

Isto pode ser entendido sobre diferentes aspectos. O primeiro deles é o volume de estruturas envolvidas, enquanto que no córtex motor os neurônios envolvidos no comando da ação motora são de menor tamanho, seus sinais também têm menor intensidade, pois as concentrações iônicas também são menores. Já nas fibras musculares, por possuírem uma quantidade maior de íons envolvidos, o sinal tem uma maior intensidade, já que um único motoneurônio é responsável pelo comando de um conjunto grande de fibras musculares (GUYTON, 1992).

Outra maneira de explicar esta diferença de potenciais tem relação com a dissipação de energia envolvida nos dois processos. Na região muscular a necessidade de energia é bem maior, pois é nessa região onde se deve realizar o trabalho de movimento e de força efetiva (GUYTON, 1992).

A figura 8 demonstra como se comporta o denominado potencial de ação que é proveniente do SNC.

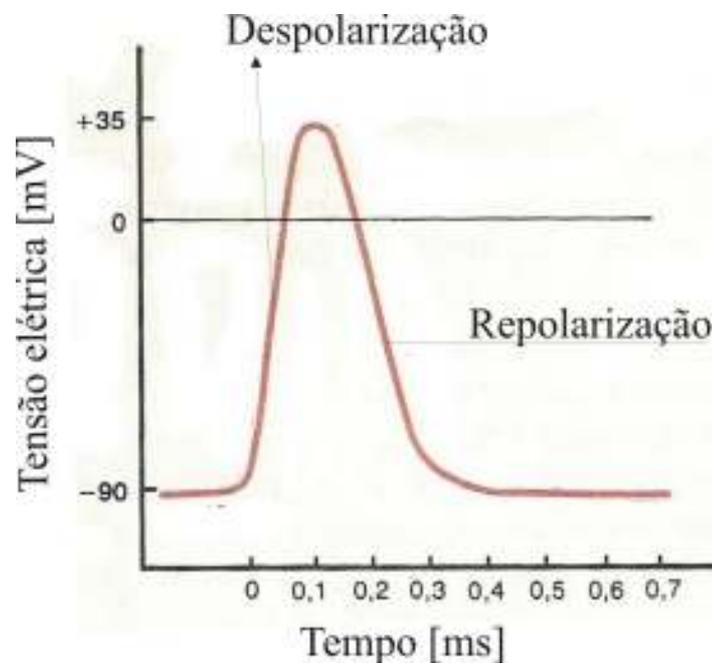


Figura 8: Potencial de ação típico (GUYTON, 1992).

2.3.2 Coleta do sinal mioelétrico

Atualmente a leitura do sinal mioelétrico é feito de duas maneiras: a primeira utiliza-se do método invasivo que introduz eletrodos em forma de agulhas diretamente nas fibras nervosas e o outro utiliza eletrodos de superfície que captam os sinais através da pele humana (DE LUCA, 2002).

A vantagem do uso de eletrodos não invasivos se deve ao fato de serem facilmente manipulados e fixados na superfície onde se encontra a maior região do músculo a ser analisado, sem causar nenhuma espécie de dano no organismo.

A desvantagem está no fato de o sinal na superfície da pele possuir menor intensidade devido à impedância da interface do corpo humano e também ao fato deste sinal estar mais suscetível a efeitos de interferências externas, como por exemplo, o sinal de 60 Hz proveniente de alimentadores elétricos externos (DE LUCA, 2002).

O modelo básico de coleta e condicionamento do sinal mioelétrico se dá com o uso de três eletrodos, através de uma leitura de sinal diferencial. Há um eletrodo que é o ativo, um segundo eletrodo de referência, ambos posicionados sobre o ventre muscular. O terceiro eletrodo é o eletrodo de aterramento, que serve para minimizar os efeitos de sinais que interferem com a leitura, que deve ser posicionado fora da região do músculo em questão (BASMAJIAN, 1962).

A figura 9 indica a posição de um determinado eletrodo sobre a região do ventre de um músculo esquelético, região onde se obtém a maior intensidade de sinal. Logo após verifica-se, na figura 10, um experimento realizado no Laboratório de Pesquisas em treinamento de atletas, localizado no estado da Pensilvânia, USA.

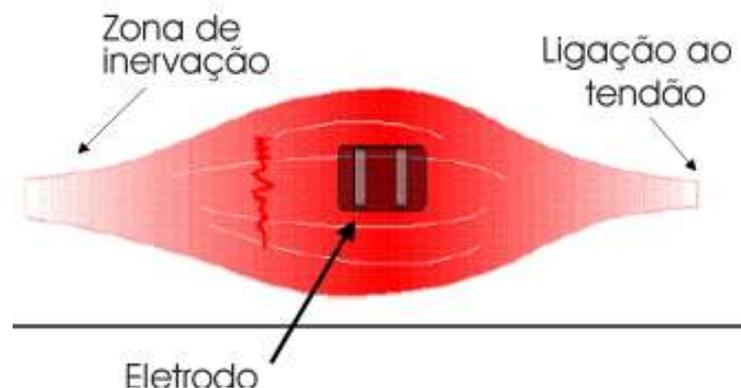


Figura 9: Imagem do ventre muscular com eletrodo posicionado (DE LUCA, 2002).

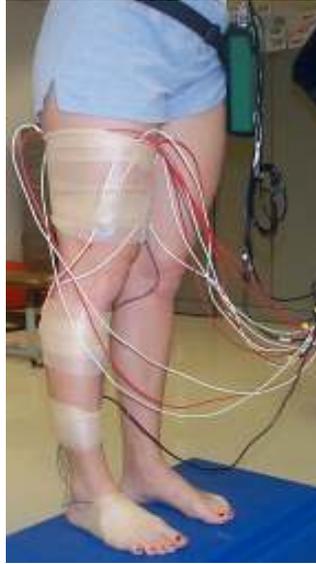


Figura 10 - Posicionamento dos eletrodos para aquisição do sinal mioelétrico (Pennsylvania State University, 2008).

2.3.3 Eletrodos

A coleta do sinal mioelétrico é feita através de eletrodos. Estes são constituídos de ligas metálicas cuja característica principal é uma condutibilidade elétrica relativamente grande (WEBSTER, 2006).

Na figura 11 pode-se observar os eletrodos do tipo não invasivos, utilizados sobre a superfície da pele nos locais onde se tem a maior região de abrangência do músculo a ser estudado. Estes eletrodos são reutilizáveis, com uma durabilidade relativamente grande, necessitando para seu correto manuseio uma devida assepsia a cada utilização.



Figura 11: Eletrodos não invasivos reutilizáveis (Cortesia Anzamed).

2.4 PROCESSAMENTO DIGITAL DO SINAL MIOLÉTRICO

O processamento digital de um sinal analógico passa pelas etapas de condicionamento, amostragem, digitalização e interpretação. Na etapa de condicionamento o sinal analógico é enquadrado dentro de uma faixa de valores máximo e mínimo para posterior amostragem (LUECKE, 2005).

Uma vez amostrado o sinal analógico, é feita sua digitalização gerando assim uma palavra binária cujo tamanho dependerá da resolução do conversor. Este valor, agora discreto no tempo, é processado através de técnicas, rotinas e funções, utilizando-se ferramentas matemáticas e de computação (LUECKE, 2005).

Na figura 12 observa-se o gráfico do valor tensão elétrica pelo tempo e na tabela à direita o valor quantizado em informação binária.

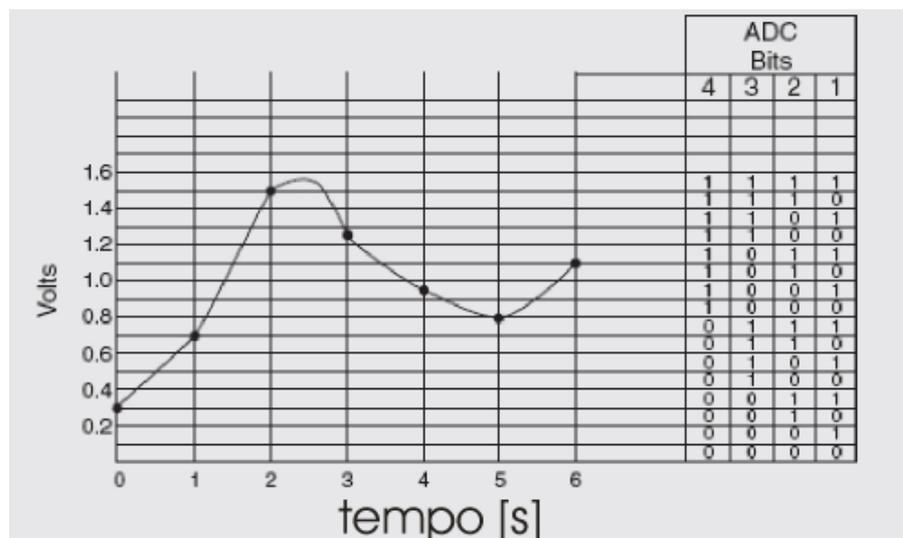


Figura 12: Exemplo de digitalização de um sinal analógico qualquer (LUECKE, 2005).

Uma das técnicas de processamento mais populares é a transformada de Fourier. Elaborada por Jean Baptiste Joseph Fourier. A transformada de Fourier determina o comportamento, no domínio da frequência, de uma função que pode ser definida analiticamente dentro de limites. A série de Fourier efetua a decomposição de um determinado sinal periódico em uma série de senos e co-senos, quando o sinal em questão não é possível ou não se consegue determinar seu comportamento analítico. Cada parcela desta soma representa as harmônicas que formam estes sinais, cada qual com sua contribuição em intensidade (amplitude) (SPIEGEL, 1977).

A equação 1 indica a transformada de Fourier (LEPPLE, 1996).

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt \quad (1)$$

Onde :

$x(t)$ é a função no domínio do tempo;

$X(f)$ é a função no domínio da frequência;

Na equação 2 pode-se verificar as séries de Fourier. A seguir, na equação 3 observa-se a forma como se determina os coeficientes a_n e na equação 4, como se determina os coeficientes b_n (SPIEGEL, 1977).

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right) \quad (2)$$

Onde :

a_0 é o coeficiente inicial;

a_n e b_n são os coeficientes a serem calculados;

L é a parcela positiva do intervalo do sinal que vai de $(-L, L)$;

n é o número de parcelas de senos e co-senos.

$$a_n = \frac{1}{L} \int_{-L}^L f(x) \cos\left(\frac{n\pi x}{L}\right) dx \quad (3)$$

$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx \quad (4)$$

Onde:

$f(x)$ é a função ou sinal a ser decomposto nas séries.

a_n e b_n são os coeficientes a serem calculados;

Para aplicações de processamento digital de sinais, onde os requisitos de tempo de processamento são críticos, utiliza-se a chamada transformada rápida de Fourier (FFT), onde

as parcelas da soma são reduzidas nas que são mais significativas e também reduzindo a quantidade de amostras, deixando mais ágil o processamento (OPPENHEIM; SCHAFER; BUCK, 1998).

Na figura 13 pode-se observar que a FFT utiliza procedimentos da DFT (Discrete Fourier Transform) para o cálculo e utiliza um arranjo entre pares de uma seqüência de amostras, no caso da figura tem-se uma amostras de 8 valores, $N = 8$. Este arranjo foi realizado pois no emprego da DFT, verificou-se a geração de repetições dos valores, visto que as funções de seno e co-seno são repetitivas em vários pontos, pelo menos duas vezes em cada ciclo. O objetivo do uso da FFT é o de determinar o espectro que freqüências que compõem o sinal mioelétrico (no caso deste projeto) e determinar que freqüências são predominantes em termos de energia do sinal.

Entrada $\{f(0), f(1), f(2), f(3), f(4), f(5), f(6), f(7)\}$.

$\{f(0), f(4), f(2), f(6), f(1), f(5), f(3), f(7)\}$.

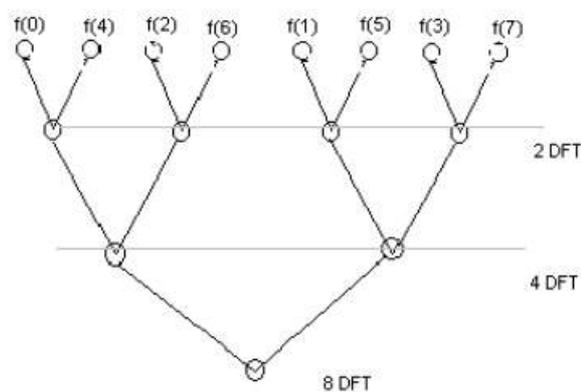


Figura 13: Organização da FFT (Universidade Federal de Pernambuco, 2008).

A transformada de Fourier, seja na forma analítica ou nas suas formas numéricas observadas na DFT e FFT, representa uma ferramenta útil na análise de sinais. Com ela é possível avaliar um determinado sinal do ponto de vista da sua composição. Podendo ser utilizado com filtro de informações em sinais de vídeo, áudio e sinais biométricos (LEPPLE, 1996).

Outras técnicas de processamento como métodos de amostragem são também empregados para auxiliar no processamento das informações. Um dos métodos é conhecido como janelamento. O objetivo deste método é o de minimizar os efeitos de possíveis

descontinuidades causadas pelo truncamento do sinal quando o mesmo é amostrado. Este método multiplica uma função ou a soma de funções com a função principal ou sinal adquirido (IFEACHOR; JERVIS, 1993).

Na equação 5 pode-se observar um dos tipos de janelamento chamado de Hamming. A função principal ou sinal adquirido em forma de um número finito de amostras é multiplicado por esta equação. Na figura 14 observa-se o formato da janela de Hamming para um valor de 60 amostras (IFEACHOR, JERVIS, 1993).

$$f(n) = 0,54 - 0,46 \cos\left(\frac{2\pi n}{N}\right) \quad (5)$$

Onde:

$f(n)$ é a função no ponto da amostra n ;

n é a amostra atual do sinal;

N é o número total de amostras.

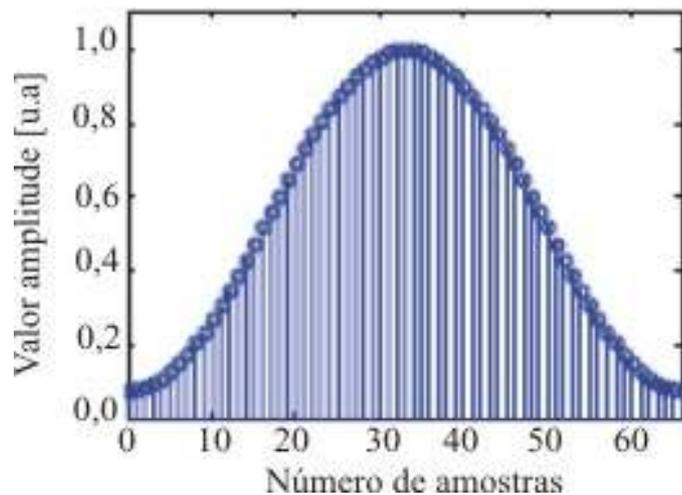


Figura 14: Exemplo de janela de Hamming (IFEACHOR; JERVIS, 1993).

O efeito sobre o sinal original é o de uma melhor conformação de pequenas transições de nível e eliminação de sinais de baixa intensidade nas proximidades (IFEACHOR; JERVIS, 1993).

3 DESENVOLVIMENTO DA BIOPRÓTESE

Este projeto pretende efetuar a leitura dos sinais elétricos de dois músculos do antebraço humano, interpretar estes sinais e atuar sobre uma prótese mecânica com o intuito de repetir estes movimentos.

O projeto do dispositivo inicia com a definição do que se pretende medir, qual a metodologia a ser utilizada para interpretar os sinais lidos e qual a maneira de expressar as informações em termos de atuação.

Os sinais a serem lidos serão provenientes de dois músculos localizados no antebraço humano. Um dos músculos é o flexor profundo dos dedos, mostrado na figura 15.A. Este músculo é responsável pela flexão de quatro dedos da mão humana exceto o polegar, conforme figura 15.B. O segundo músculo é o flexor ulnar do carpo, mostrado na figura 16.A. Este músculo atua no movimento de adução da mão, ou movimento radial, conforme mostrado na figura 16.B.

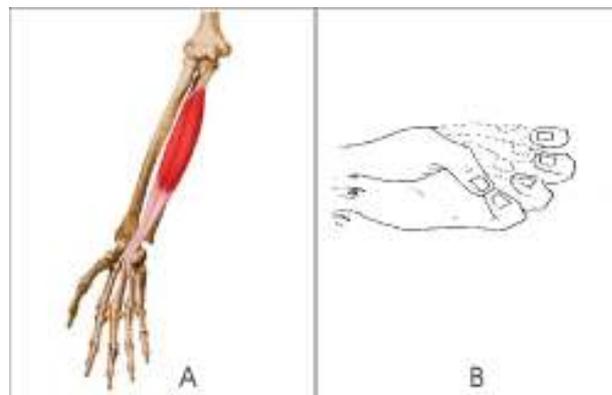


Figura 15.A: Músculo flexor profundo dos dedos – Figura 15.B : Movimento de flexão dos dedos.

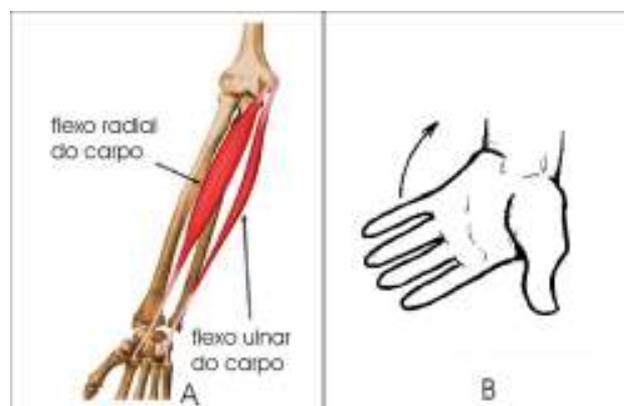


Figura 16.A: Músculo flexor ulnar do carpo – Figura 16.B : Movimento de adução da mão.

Posteriormente é feita a aquisição dos sinais, sua digitalização e interpretação. Para interpretar os sinais optou-se por aplicar uma FFT na seqüência de amostras. Com isso pode-se analisar qual a frequência do sinal é predominante no comando dos músculos. Com a localização deste valor é possível comparar a intensidade do sinal neste ponto e efetuar a comparação com um valor previamente estipulado.

Para expressar os resultados obtidos das análises utilizaram-se gráficos e a atuação em um dispositivo mecânico com dois motores de passo cada qual responsável por expressar a ação de cada movimento lido.

Na figura 17 verifica-se o diagrama em blocos de toda a estrutura. No item 3.1 serão detalhadas cada parte com suas características e componentes envolvidos. Como pode ser observado no diagrama o sinal mioelétrico é captado pelos eletrodos posicionados na superfície da pele na região sobre o ventre muscular. O sinal captado é então condicionado, digitalizado e enviado serialmente pela interface de aquisição até a unidade central. A unidade central por sua vez efetua o processamento do sinal separando-o nas suas componentes espectrais. Cada componente do sinal contribui com sua parcela de energia e este valor, em termos percentuais, é registrado em uma tabela montada pela rotina da FFT. Pela observação experimental, avalia se quais as frequências que contribuem com maior parcela e é nestes pontos em que a unidade central efetua as comparações. Esta comparação é feita por um valor previamente fornecido pelo operador que pode ajustá-lo de acordo com sua observação do comportamento do sinal naquele ponto. Quando o sinal lido excede este valor de comparação, a unidade central envia um comando para a interface de potência para que seja ativado o motor correspondente ao movimento detectado. Se o sinal é inferior a este valor de comparação, a unidade central envia outro comando para que o motor retorne para seu ponto de partida.

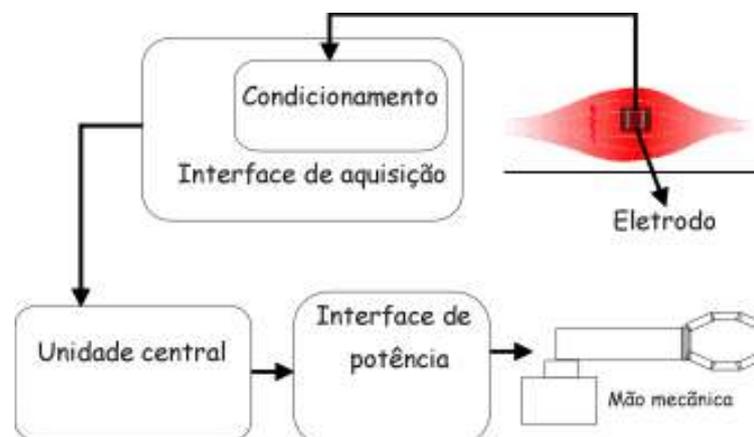


Figura 17: Diagrama em blocos do projeto.

3.1 ELETRODOS E POSICIONAMENTO

Os eletrodos utilizados são do tipo não-invasivos, reutilizáveis e formados por uma liga metálica de níquel e cobre. Optou-se por este tipo de eletrodo devido a sua relativa durabilidade, resistência mecânica e boa condutibilidade elétrica. Na figura 18 pode-se verificar, em detalhes, duas amostras destes eletrodos.

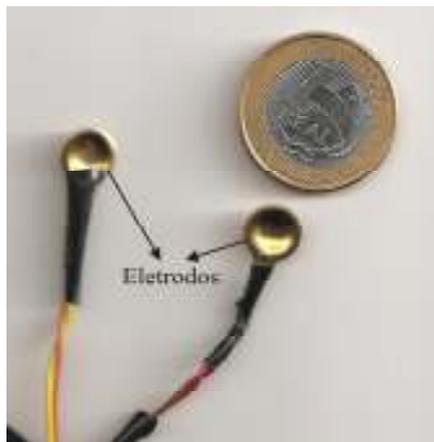


Figura 18: Um par de eletrodos.

Neste projeto são utilizados cinco eletrodos sendo que para cada músculo lido é usado um par e o quinto eletrodo é a referência. Este eletrodo de referência é fixado no lóbulo da orelha e interligado no ponto comum da placa de aquisição.

Cada par de eletrodos é posicionado em dois pontos sobre a pele acima do músculo que se deseja medir, conforme a figura 19. O cabo de interligação é formado por um par de condutores trançados e com blindagem para minimizar os efeitos de interferências externas. Os eletrodos são fixados sobre o ventre muscular, ponto onde se obtém uma melhor captação do sinal mioelétrico.

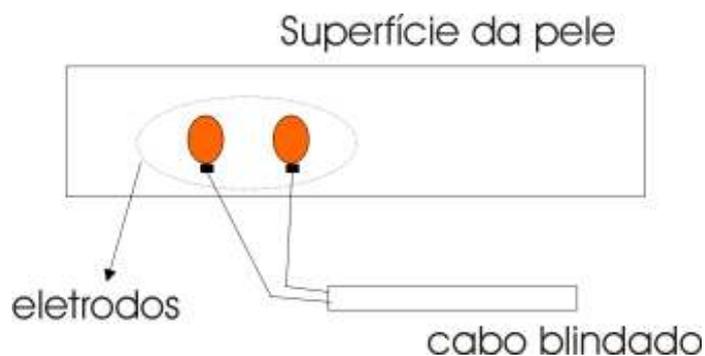


Figura 19: Eletrodos posicionados sobre a pele.

Na figura 20.A pode-se observar o posicionamento dos eletrodos sobre o músculo flexor profundo dos dedos e na figura 20.B os eletrodos sobre o músculo flexor ulnar do carpo.



Figura 20. (A): Eletrodos posicionados sobre o músculo flexor profundo dos dedos. (B): Eletrodos sobre o músculo flexor ulnar do carpo.

3.2 CONDICIONAMENTO E INTERFACE DE AQUISIÇÃO

O sinal coletado pelos eletrodos chega à etapa de condicionamento formado por amplificadores operacionais de instrumentação e de uso geral. Nesta etapa é feita a adequação deste sinal analógico dentro da faixa de valores de tensão de entrada do conversor AD.

O circuito eletrônico da etapa de condicionamento é formado por um amplificador de instrumentação de referência INA 101, fabricante Texas Instruments, e por dois estágios com dois amplificadores de uso geral um na configuração de subtrator e outro como amplificador inversor, referência OPA 4227, fabricante Texas Instruments. Este circuito constitui um canal de entrada para o sinal do músculo. Este circuito é alimentado por duas baterias seladas de 6V com uma capacidade de 1,3 Ah e que se encontram ligadas em série. O ponto comum entre estas baterias é ligado ao referencial do circuito analógico fazendo com que se possa obter tensões simétricas de +6V e -6V para alimentar os amplificadores operacionais. Este circuito tem um consumo médio de 20mA que possibilita uma autonomia das baterias de cerca de 65 horas.

Na figura 21 pode-se observar o circuito eletrônico de condicionamento de um canal de entrada analógico em formato de diagrama de blocos. Neste circuito o amplificador de instrumentação proporciona um ganho diferencial entre os sinais dos eletrodos. Esta diferença amplificada passa por um bloco subtrator inversor, cuja função é realizar um deslocamento no sinal DC e fazer com o sinal possa ocupar apenas o primeiro quadrante (somente positivo). Neste caso o ponto “A” indica a entrada do valor a ser subtraído. A etapa seguinte é composta por um amplificador inversor cuja função é inverter novamente o sinal e proporcionar mais um ganho ajustável no valor de X.

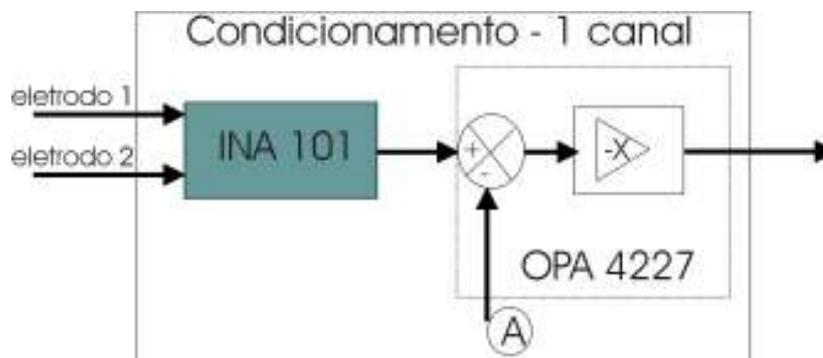


Figura 21: Diagrama do condicionador para um canal.

O sinal que sai do condicionamento já está ajustado para ser lido pela entrada analógica do conversor AD que vai de 0,00 a 5,00V. Este conversor faz parte de um dos módulos do microcontrolador utilizado, no caso o PIC 16F72, do fabricante Microchip. Neste projeto o AD está configurado para uma taxa de aquisição de aproximadamente 17.000 amostras por segundo.

A etapa de aquisição então, é composta pelo microcontrolador executando um programa que efetua sucessivas leituras da porta analógica. Os valores lidos são convertidos para uma palavra binária com resolução de 8 bits, ou seja, o valor do sinal analógico é quantizado numa faixa de 256 valores. Estes valores digitalizados são enviados, serialmente, para a unidade central utilizando o protocolo padrão RS-232 em uma velocidade de 19.200 bps (bits por segundo). A escolha do microcontrolador se deu pelo seu bom desempenho para as funções que se propõem, pelo baixo custo e fácil aquisição no mercado.

Na figura 22 pode-se observar o diagrama em blocos da etapa de aquisição do sinal. Nela verifica-se que a interface utilizada para envio pelo protocolo RS-232, é provida pelo componente HIN 232, fabricante Intersil. Este diagrama representa a estrutura para aquisição do sinal de um músculo. A mesma se repete para o segundo sinal exceto a unidade central que

é comum. Este circuito é alimentado por uma bateria de 12V e 1,3 Ah e possui um consumo médio de 18mA. Estas baterias também suprem energia para a etapa de potência e motores cujo consumo médio é de 100mA. Neste caso o total de consumo dos dois módulos será de 118mA fazendo com que a autonomia seja de aproximadamente 11 horas. As baterias são do tipo seladas e recarregáveis.

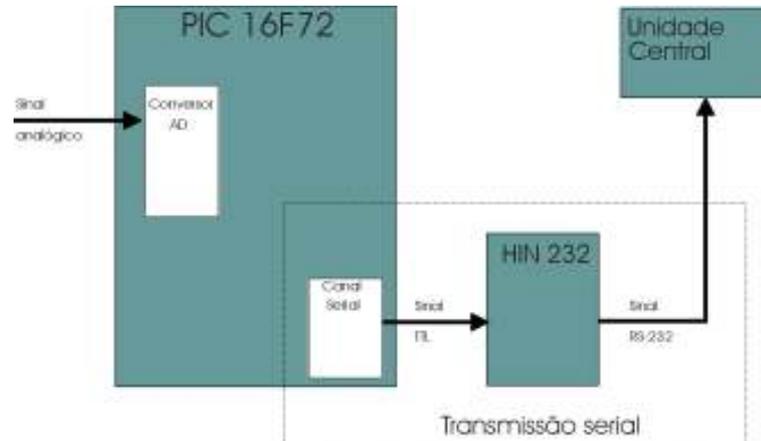


Figura 22: Interface de aquisição para um canal.

O programa executado no microcontrolador foi desenvolvido em linguagem C e o seu fluxograma pode ser visto na figura 23. A seqüência básica de funcionamento é a de criação de variáveis, ativação do canal do conversor AD e envio do valor convertido, através do canal serial, para a unidade central. Sendo que o laço de repetição ocorre apenas no conversor AD e no envio dos valores pela serial.



Figura 23: Fluxograma do programa executado no PIC16F72.

3.3 UNIDADE CENTRAL

A unidade central é composta por um computador pessoal padrão executando um aplicativo dedicado a receber os sinais da interface de aquisição, interpretar e atuar sobre a interface de potência.

O diagrama em blocos da unidade central pode ser visto na figura 24. Nela pode-se verificar que cada bloco representa uma rotina de processamento.

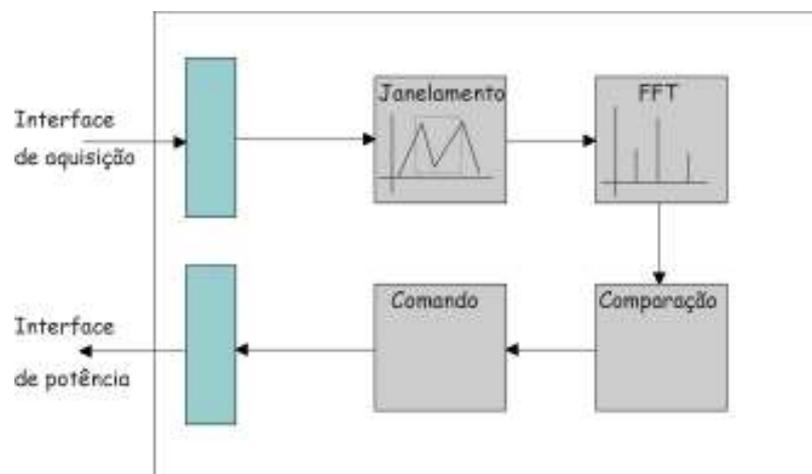


Figura 24: Diagrama em blocos da unidade central.

No diagrama da figura 24 o sinal chega pelo canal serial onde é montada uma tabela com 32 valores de amostragem do sinal analógico. Esta tabela é submetida a uma rotina de janelamento que pode ser definida pelo usuário. Após o sinal passa por uma FFT onde se obtém o espectro de frequências mais significativas que formam o sinal original. Feito o desmembramento do sinal efetua-se a comparação por nível de tensão exatamente nos pontos específicos das frequências predominantes, no caso, na faixa entre 40 e 45 Hz para o músculo flexor profundo dos dedos e 75 e 80 Hz para o músculo flexor ulnar do carpo. Esta ação é feita pelo bloco de comparação que tem como sinal-base um valor pré-determinado pelo usuário. Caso o sinal exceda em amplitude dos valores para aqueles pontos de frequência o bloco de comando se encarrega de enviar um código para a interface de potência comandando o respectivo motor responsável pelo movimento em questão.

Na figura 25 pode-se ver o fluxograma do aplicativo que é executado na unidade central. Este fluxograma representa a rotina de avaliação de apenas um movimento, determinado como movimento X. Sendo que o outro movimento segue a mesma seqüência de procedimentos. Nele observa-se que a unidade central fica, constantemente, avaliando os sinais mioelétricos e enviando comandos, para a interface de potência, para avançar ou recuar o motor do respectivo movimento.

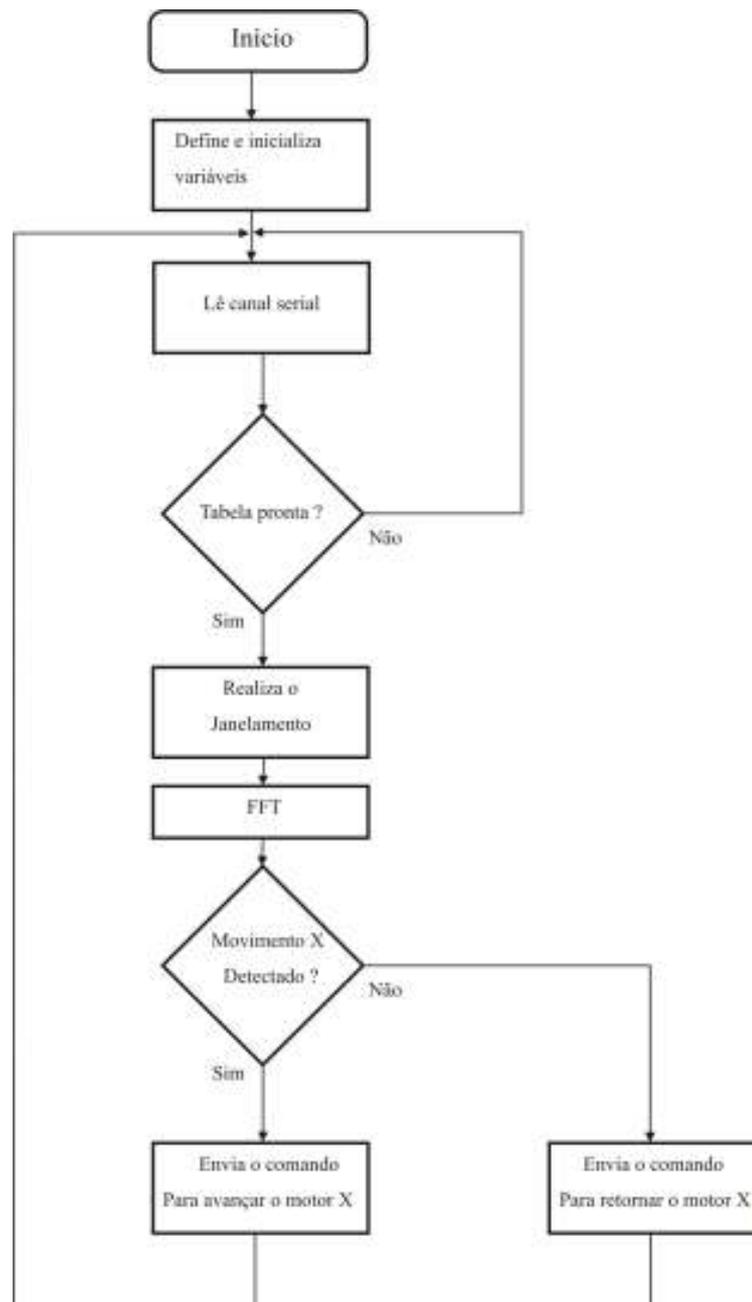


Figura 25: Fluxograma da unidade central.

Na figura 26 pode-se observar a tela principal do programa que roda na unidade central. Este aplicativo foi desenvolvido em linguagem padrão ANSI-C e C++ utilizando o ambiente de programação da empresa Borland o Builder versão 5.0.

Na figura 26 é possível verificar a presença de três gráficos. O gráfico G1 indica o sinal analógico digitalizado proveniente do músculo a ser medido, ou seja, o sinal em tempo real da medição. O eixo das abscissas indica a quantidade de amostras total depois de uma determinada seqüência de ciclos de leitura. O eixo das ordenadas indica o valor quantizado do sinal da faixa de 0 à 5 Volts para a faixa de 0 à 255 em valor decimal.

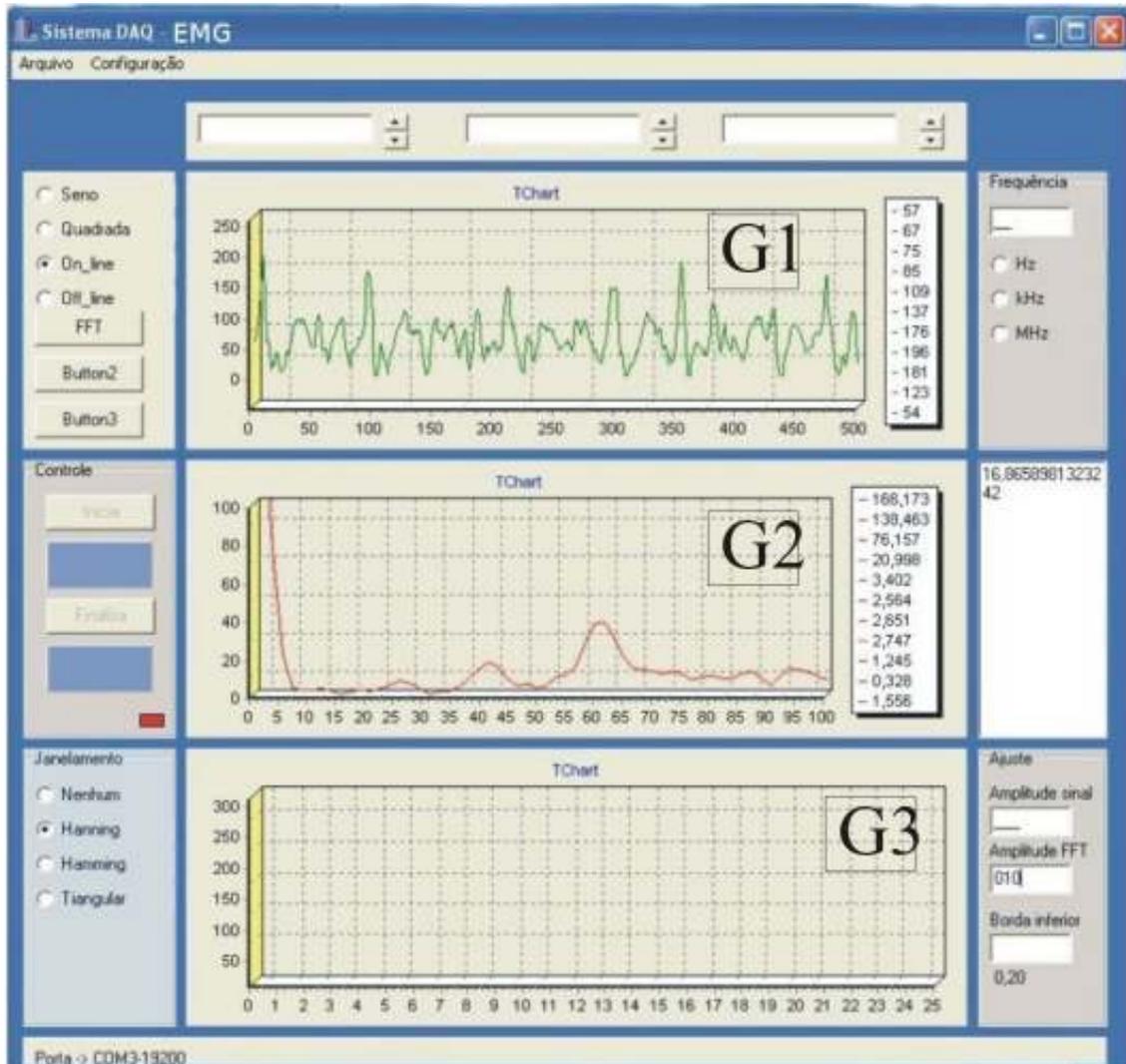


Figura 26: Tela principal do programa.

O gráfico G2 indica o sinal processado pelas rotinas de janelamento e pela FFT. O eixo das abscissas indica o valor em freqüência com a unidade Hertz e o eixo das ordenadas indica o valor em amplitude quantizada de 0 à 255.

O gráfico G3 indicará a mesma situação do gráfico G1 só que para sinal proveniente do segundo músculo a ser analisado. Esta etapa da rotina ainda não se encontra implementada.

3.4 INTERFACE DE POTÊNCIA E MÃO MECÂNICA

A interface de potência é composta por dois microcontroladores PIC 16F72, componentes que fazem a interface com os motores e circuitos condicionadores para os sensores de fim de curso. Todo o sistema é alimentado por uma bateria selada recarregável de 12V / 1,3 Ah.

O mesmo conceito adotado na etapa de aquisição se repete na etapa de potência onde cada microcontrolador efetua o acionamento e controle de um determinado movimento, isto proporciona maior rapidez no atendimento das funções. Cada microcontrolador fica aguardando instruções da unidade central. Estas instruções vêm através do mesmo canal serial que a unidade central recebe informações da interface de aquisição. Cada unidade microcontrolada da etapa de potência então, recebe um comando específico para cada movimento que foi detectado.

Na figura 27 pode-se verificar o diagrama em blocos da etapa de potência. Este diagrama é relativo ao acionamento de um motor sendo que a mesma estrutura se repete para o acionamento do segundo movimento.

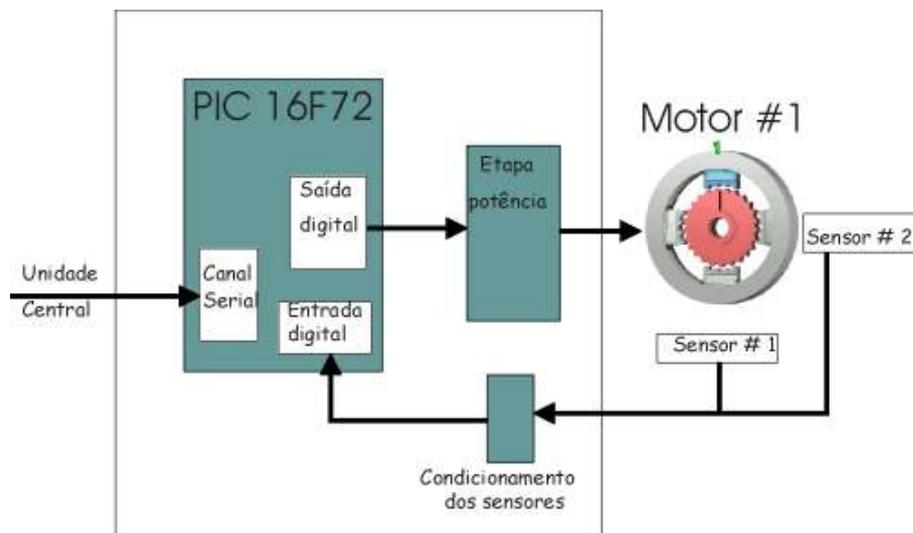


Figura 27: Diagrama etapa de potência.

Na figura 28 verifica-se o fluxograma do código que roda no microcontrolador pertencente à etapa de potência. Nele é possível verificar que o programa se mantém em um laço repetitivo verificando o canal serial à espera de um comando válido enviado pela unidade central. Após a recepção, é verificado a qual motor se destina o comando, e o controle é feito

pelas rotinas de cada movimento (radial ou flexão). As rotinas de controle dos movimentos também fazem o monitoramento dos sensores que delimitam o fim de cursos de cada atuação.

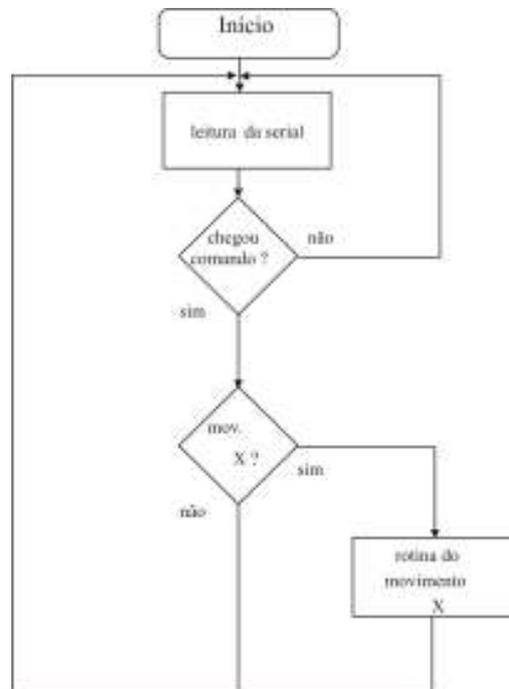


Figura 28: Fluxograma etapa de potência.

O mecanismo que compõe a mão mecânica é formado por dois motores de passo de pequeno porte. O motor denominado motor flexão é responsável por representar o movimento de flexão profunda dos dedos. O motor denominado motor radial representa o movimento de flexão ulnar do carpo.

O motor radial possui ainda dois sensores laterais para limitar o movimento de rotação. Cada sensor é ativado por pequenas chapas metálicas presas a estrutura da haste que efetua o giro. Esta estrutura é apoiada lateralmente por duas roldanas (uma de cada lado) que facilitam a rotação do conjunto. Já o motor de flexão apenas gira livremente servindo para indicar que o movimento de flexão dos dedos foi reconhecido pelo sistema. Quando o movimento dos músculos é detectado e avaliado pela unidade central, um comando é enviado para a interface de potência que por sua vez realiza o acionamento do respectivo motor.

Se o motor acionado for responsável pelo movimento radial, então este efetua um movimento de saída do ponto inicial até atingir o ponto final onde ativa o sensor de fim de curso. Se o motor acionado for o responsável pelo movimento de flexão profunda dos dedos, então a interface de potência aciona o motor durante um tempo de aproximadamente 5 segundos para um sentido ou para outro dependendo do comando recebido.

A figura 29 demonstra o projeto gráfico da mão mecânica.

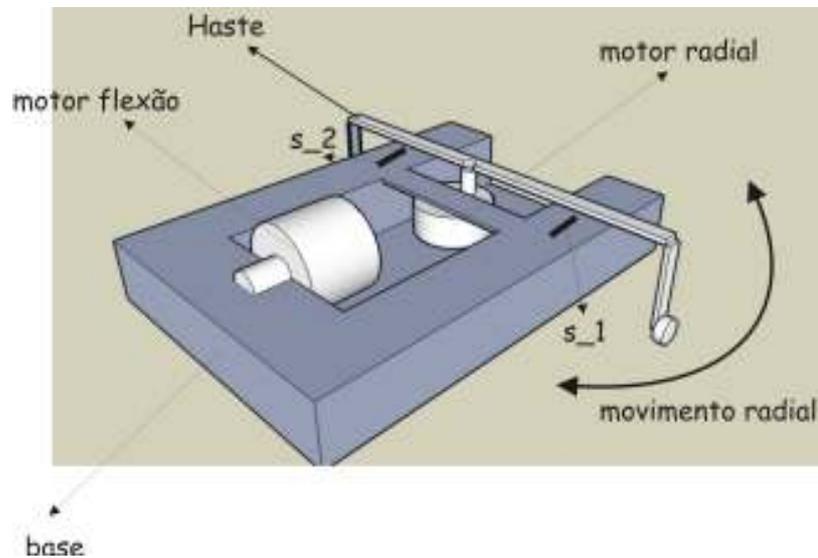


Figura 29: Projeto gráfico da mão mecânica.

Na figura 29 pode-se observar os componentes básicos do mecanismo responsável por emular o movimento da mão humana. Este dispositivo também auxilia em demonstrar externamente a resposta ao reconhecimento dos padrões do sinal mioelétrico. Os pontos indicados por s_1 e s_2 correspondem aos sensores limites para os movimentos no sentido horário e anti-horário respectivamente.

Na figura 30 pode-se observar a foto tirada do dispositivo mecânico montado. Nela se verifica os mesmos componentes indicados no projeto gráfico da figura 29. Neste dispositivo o motor radial executa o movimento da haste no sentido horário ou anti-horário, de acordo com a presença ou não do movimento do músculo flexor profundo dos dedos. Na haste existem, em cada extremidade, um anteparo que serve para interromper os sensores fotoelétricos fixados na base. Estes sensores servem para limitar a atuação do movimento de maneira que o motor não fique girando continuamente. O programa da interface de potência se encarrega de ler estes sinais toda a vez que o motor for acionado. Na figura 31 é possível verificar um modelo de como o anteparo preso na haste interrompe o sinal luminoso do foto-emissor fazendo com seja ativada a saída digital do foto-sensor.

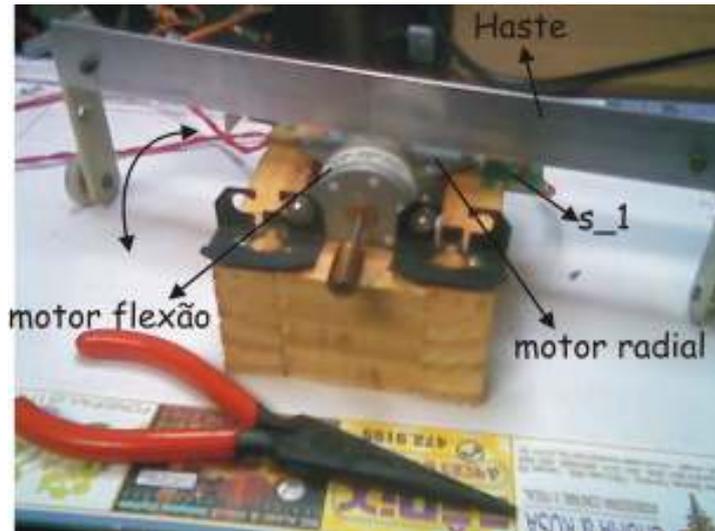


Figura 30: Mão mecânica montada.

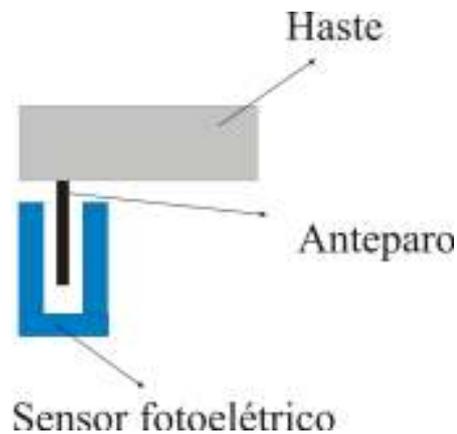


Figura 31: Anteparo preso na haste interrompendo o sinal do sensor fotoelétrico.

3.5 DISPOSITIVO AUXILIAR

Durante o desenvolvendo do projeto foi possível implementar um colete que possibilita a integração das partes integrante em um único invólucro. Este possui compartimentos na parte traseira permitindo alojar as baterias que suprem energia para os sistemas e as placas de controle.

Pela figura 32.A pode-se verificar o colete com o alojamento superior reservado para as placas de controle e na parte inferior as fontes de energia. Na figura 32.B verifica-se o colete em uso com os cabos para os eletrodos.

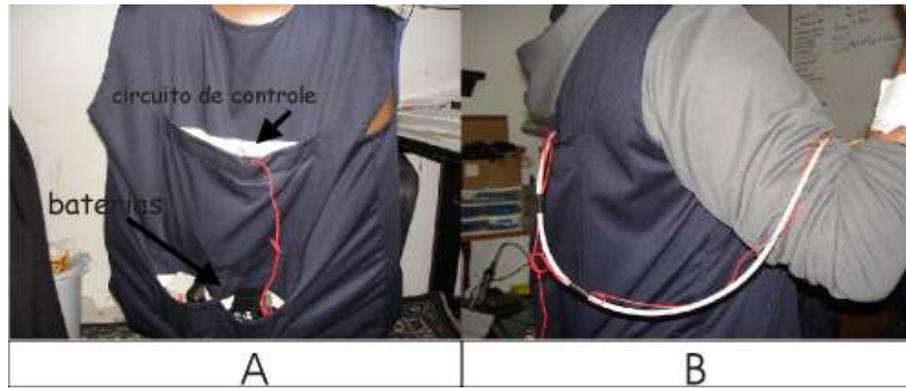


Figura 32.A: Colete vista traseira. 32.B: Colete vista lateral sendo vestido pelo usuário.

4 RESULTADOS E DISCUSSÕES

Neste capítulo são apresentados os resultados das avaliações feitas no protótipo que visam a validação do mesmo. Os testes foram desenvolvidos para efetuar a validação das partes integrantes em separado, dos sinais mioelétricos e avaliação do conjunto interligado.

4.1 TESTES FUNCIONAIS DAS PARTES

Os testes funcionais começam com a avaliação do sistema de condicionamento. O procedimento para análise consiste em aplicar um sinal na entrada, da etapa de condicionamento, e avaliar o valor da saída. A figura 33 demonstra um diagrama em bloco do sistema de testes. O objetivo deste teste é o de avaliar o funcionamento da etapa de condicionamento.

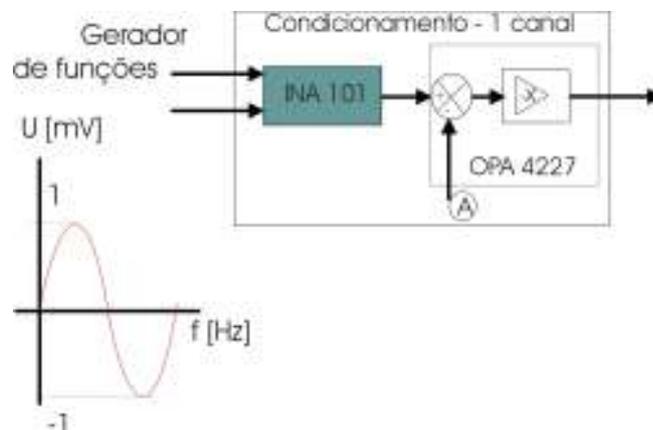


Figura 33: Diagrama de teste da etapa de condicionamento.

Foi aplicado um sinal senoidal com uma amplitude de aproximadamente 1 mV de pico numa frequência de 1kHz. A amplitude do sinal observado na saída do condicionamento apresentou um valor de aproximadamente 800 mV de pico. O amplificador de instrumentação INA 101 foi implementado para proporcionar um ganho 400 vezes. O segundo estágio de amplificação fica a cargo do amplificador OPA4227 que teve seu ganho ajustado para um valor de 2 vezes. Neste segundo estágio ainda existe um entrada, indicada pela letra “A”, onde

é possível, com o uso de um potenciômetro, ajustar o off-set da saída final do condicionamento.

Outro teste funcional se deu na avaliação do sistema de aquisição e transmissão serial de dados. Este sistema é baseado no uso de um microcontrolador PIC16F72 que converte os valores da etapa de condicionamento e os envia pelo canal serial. A conversão é feita no canal 0 deste microcontrolador cuja resolução é de 8 bits. A transmissão das informações é feita para a unidade central numa taxa de 19200 bps (bits por segundo).

O teste desta etapa teve o objetivo de avaliar o erro na conversão dos sinais aplicados no canal 0 do conversor AD do microcontrolador. Para isto foi aplicado um sinal ajustado em 5 pontos dentro da faixa de 0,00 Volts a 5,00 Volts. O resultado do experimento pode ser observado na tabela 1 e tabela 2, onde a primeira coluna representa os pontos da faixa onde se realizaram as medidas, a segunda coluna indica a tensão aplicada à entrada e a terceira coluna indica o valor da conversão em valores decimais. Neste experimento os valores foram sendo aplicados em 5 pontos da faixa primeiramente em sentido crescente de valor e após em no sentido decrescente. Nos gráficos das figuras 34.A e 34.B pode-se verificar o comportamento do conversor face aos valores colocado em sua entrada.

Tabela 1: Ensaio na conversão sentido crescente

Ponto de teste [%]	Valor de tensão elétrica [V]	Valor esperado	Valor convertido
0	0,00	0	1
25	1,25	64	64
50	2,50	128	128
75	3,75	191	192
100	5,00	255	254

Tabela 2: Ensaio na conversão sentido decrescente

Ponto de teste [%]	Valor de tensão elétrica [V]	Valor esperado	Valor convertido
100	0,00	255	254
75	1,25	191	190
50	2,50	128	128
25	3,75	64	63
0	5,00	0	1

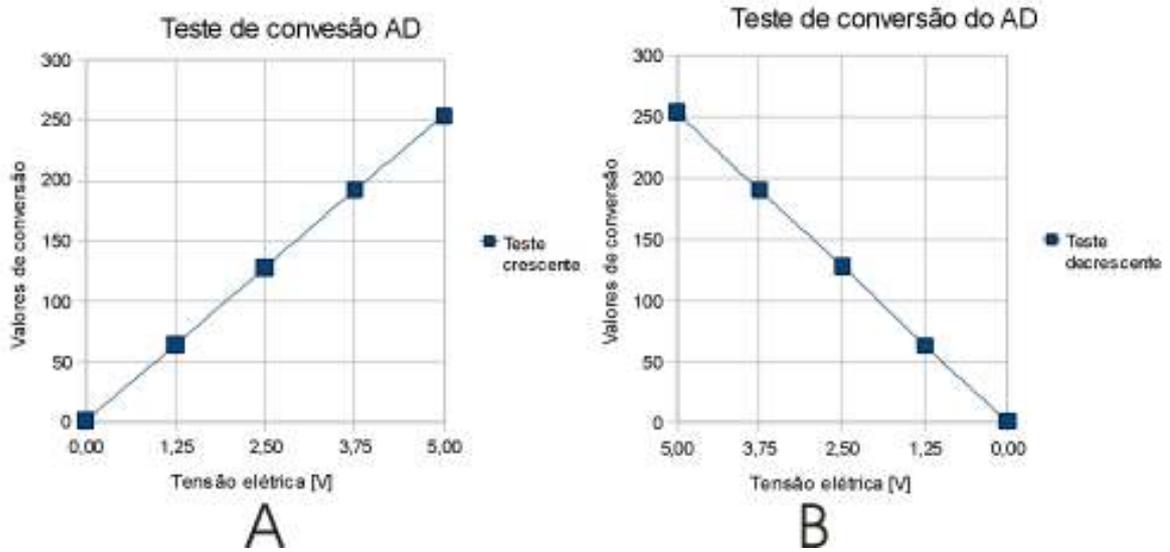


Figura 34.A: Teste do conversor AD sentido crescente. Figura 34.B: Teste do conversor AD sentido decrescente.

O erro máximo, em unidades de conversão, apresentado no sentido crescente e decrescente foi de uma unidade o que representa 0,4% do valor máximo da conversão. Este valor pode ser desprezado uma vez que se equivale ao valor da resolução do AD que é de 0,02V. De acordo com estes resultados é possível concluir que a etapa de aquisição opera dentro de uma margem de erros aceitáveis.

Avaliou-se também o aplicativo desenvolvido para a rotina da FFT. O procedimento consistiu na aplicação de um sinal senoidal em três valores de frequências diferentes. O objetivo é uma análise qualitativa da FFT, verificando se a rotina atende às requisições. Nos gráficos das figuras 35, 36 e 37 estão às avaliações realizadas com uma frequência de 10 Hz, 50 Hz e 90 Hz respectivamente.

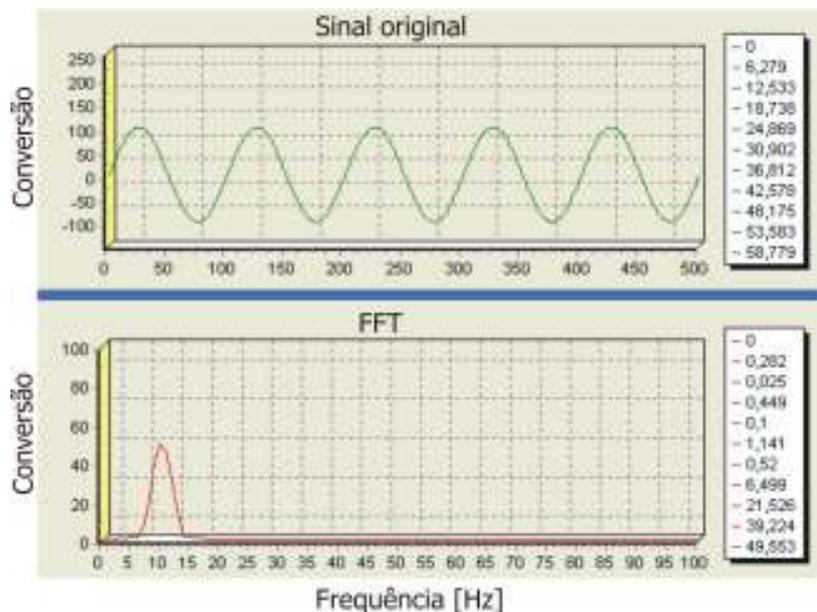


Figura 35: Teste do aplicativo para sinal de 10 Hz.

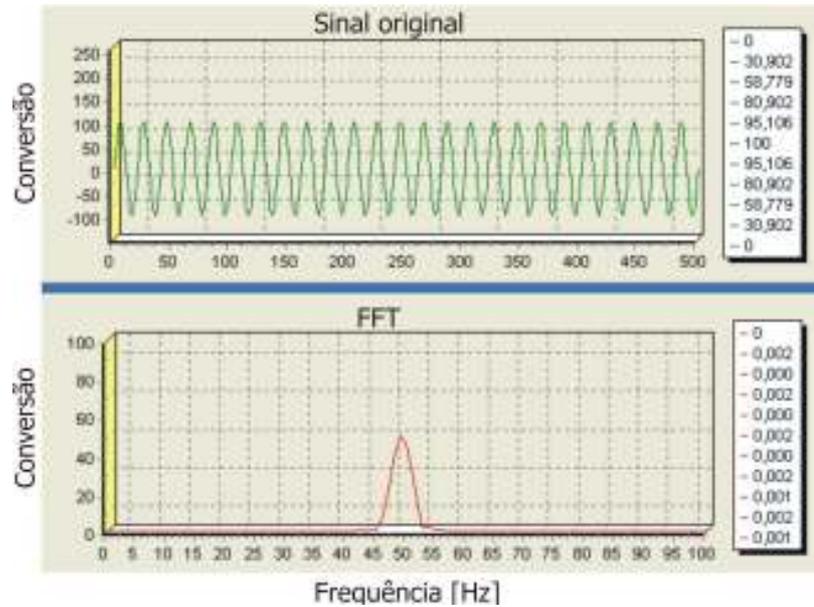


Figura 36: Teste do aplicativo para sinal de 50 Hz.

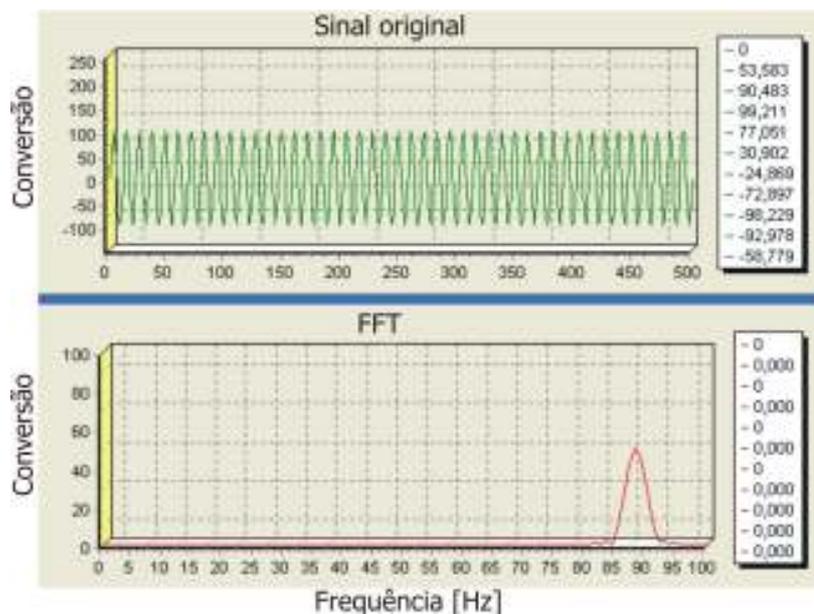


Figura 37: Teste do aplicativo para sinal de 90 Hz.

Avaliação desta rotina foi feita de maneira visual, ou seja, é verificado se no gráfico da FFT o valor da intensidade de energia está centrada na frequência do sinal aplicado. Os resultados obtidos indicam um erro máximo de 1Hz para cada frequência do sinal. Com estes resultados pode-se concluir que as rotinas matemáticas apresentam uma aproximação confiável para a análise. Ressaltando a necessidade de constante aperfeiçoamento nas rotinas como também na melhoria da resolução dos conversores AD.

O próximo teste funcional foi realizado na interface de potência e na mão mecânica. Os objetivos dos testes são validar se a interface de potência está atuando sobre os motores da mão mecânica e se esta interface recebe os comandos da unidade central.

Para realizar estes testes foi desenvolvida uma rotina de programa para ser executada no microcontrolador da etapa de potência. Esta rotina, cujo fluxograma se encontra no apêndice H, efetua o acionamento do motor que realiza o movimento radial, do motor do movimento flexor dos dedos e também testa os sensores de fim de curso laterais que limitam o movimento. Esta rotina aguarda um comando da unidade central. Se o comando indica o acionamento do motor do movimento radial em um determinado sentido, o motor é acionado caso ainda não esteja no limite. Se o motor atinge o limite em um dos sensores o motor é desativado e a rotina volta para aguardar novo comando. Se o comando for para acionar o motor de flexão dos dedos em um dos sentidos este motor é então acionado pelo tempo de 5 segundos e depois desativado e a rotina volta a aguardar um novo comando. O código fonte desta rotina de teste se encontra no apêndice I.

A avaliação de desempenho deste teste é feito visualmente, onde é verificado se os motores estão realizando os movimentos nos sentidos desejados e no tempo aproximado. Ao final destes testes foi possível concluir que a etapa de potência e os motores obedecem aos comandos da unidade central. Ressalta-se a necessidade de uma melhoria no que diz respeito a tempos de acionamento e nas rotinas de comando de maneira que se possa obter um melhor desempenho em termos dos tempos de resposta.

4.2 TESTES DOS SINAIS MIOELÉTRICOS

O outro procedimento de avaliação consistiu no teste de validação dos sinais obtidos à partir dos músculos. Para isto foram feitas leituras utilizando 3 indivíduos de faixas etárias diferentes e executando 10 ensaios de leitura para cada músculo. Os objetivos destes ensaios são avaliar a reprodutibilidade dos sinais no mesmo indivíduo e a variabilidade dos sinais entre os 3 indivíduos escolhidos.

O primeiro músculo a ser lido foi o flexor profundo dos dedos. Para isso foi posicionado um eletrodo cerca de 80 mm abaixo do pulso na região anterior do antebraço direito e o segundo eletrodo cerca de 30 mm abaixo deste primeiro. Um eletrodo de referência foi preso no lóbulo da orelha direita. A figura 38 ilustra o posicionamento dos eletrodos, os ensaios e a resposta gráfica. Nestes ensaios o indivíduo flexionava os dedos da mão direita e o aplicativo fazia o registro do percentual de energia do sinal em três pontos na faixa de

freqüência entre 40 e 45 Hz. Os três pontos medidos nos ensaios estão na freqüência de 40Hz, 42Hz e 45Hz.

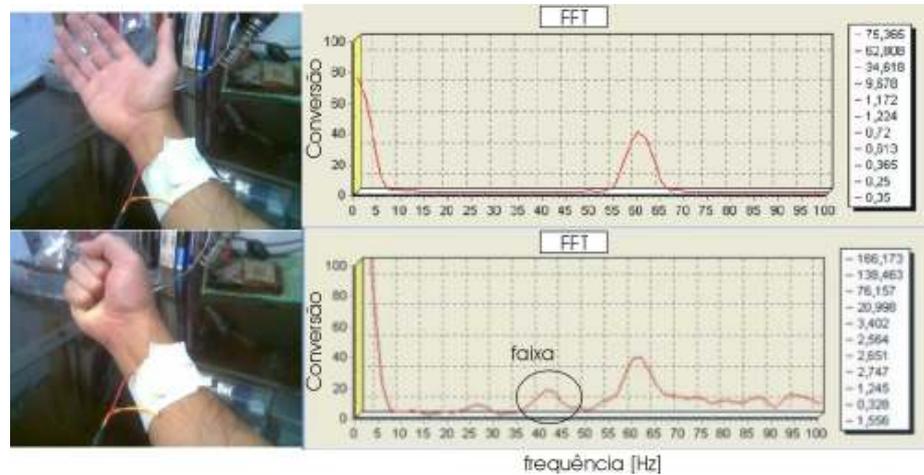


Figura 38: Imagem do ensaio no músculo flexor profundo dos dedos.

A tabela 3 indica os valores obtidos do primeiro indivíduo de idade 33 anos. Nesta tabela a primeira coluna indica o número do ensaio realizado, a segunda coluna indica o valor da quantização de energia do sinal obtido na freqüência inicial da faixa (40Hz), a terceira coluna o valor para a freqüência de 42Hz e a quarta para o valor de 45 Hz. Todos os valores são obtidos através da rotina numérica que avalia o maior valor apresentado no ponto específico durante o ensaio.

Tabela 3: Ensaio de medição da flexão profunda dos dedos no indivíduo #1

Número do ensaio	Quantização da energia do sinal na freqüência de 40 Hz[%]	Quantização da energia do sinal na freqüência de 42 Hz[%]	Quantização da energia do sinal na freqüência de 45 Hz[%]
1	15	20	13
2	15	21	13
3	15	20	14
4	16	22	14
5	14	21	13
6	15	21	14
7	15	20	14
8	14	21	13
9	14	21	13
10	15	22	14

O gráfico da figura 39 demonstra o comportamento do sinal mioelétrico nas três frequências avaliadas.

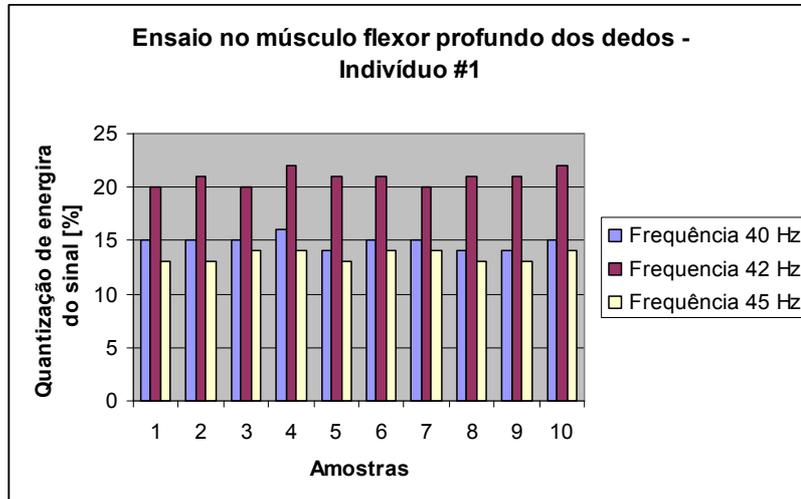


Figura 39: Gráfico dos ensaios no músculo flexor dos dedos no indivíduo #1.

O segundo músculo a ser avaliado foi o flexor ulnar do carpo. O procedimento consiste em realizar ensaios fazendo com que o indivíduo realizasse movimentos de adução mantendo a mão posicionada na máxima excursão radial durante um período de 10 segundos. Neste ensaio os eletrodos são posicionados na parte posterior do antebraço, o primeiro eletrodo cerca de 60 mm acima do cotovelo e o segundo cerca de 30 mm acima deste primeiro e o eletrodo de referência também posicionado no lóbulo da orelha. Foram realizados os mesmos ensaios nos demais indivíduos e obteve-se uma faixa de frequência mais predominante para o sinal na faixa de 75 à 80 Hz. A figura 40 demonstra o ensaio realizado.

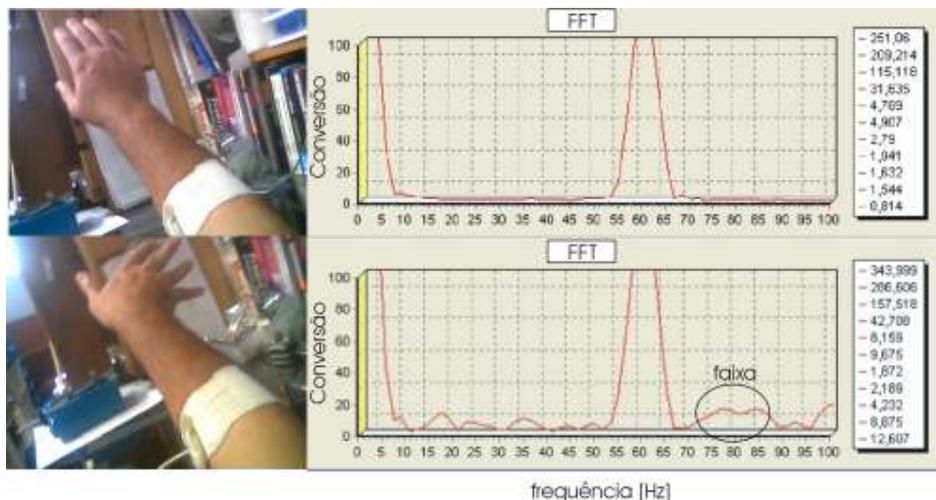


Figura 40: Imagem do ensaio no músculo flexor ulnar do carpo.

A tabela 4 indica os ensaios no mesmo indivíduo de 33 anos agora, medindo-se os sinais do músculo flexor ulnar do carpo. A estrutura da tabela é idêntica a anterior, onde a segunda coluna representa os sinais na frequência de 75 Hz, a terceira coluna na frequência média de 77 Hz e a quarta coluna na frequência de 80 Hz.

Tabela 4: Ensaio de medição da flexão ulnar do carpo no indivíduo #1

Número do ensaio	Quantização da energia do sinal na frequência de 75 Hz[%]	Quantização da energia do sinal na frequência de 77 Hz[%]	Valor de tensão elétrica obtido para a frequência de 80 Hz [mV]
1	10	11	9
2	11	12	10
3	10	12	10
4	9	10	9
5	11	12	9
6	11	13	10
7	10	11	10
8	10	11	9
9	9	10	9
10	9	11	8

O gráfico da figura 41 demonstra o comportamento do sinal nas 3 frequências avaliadas.

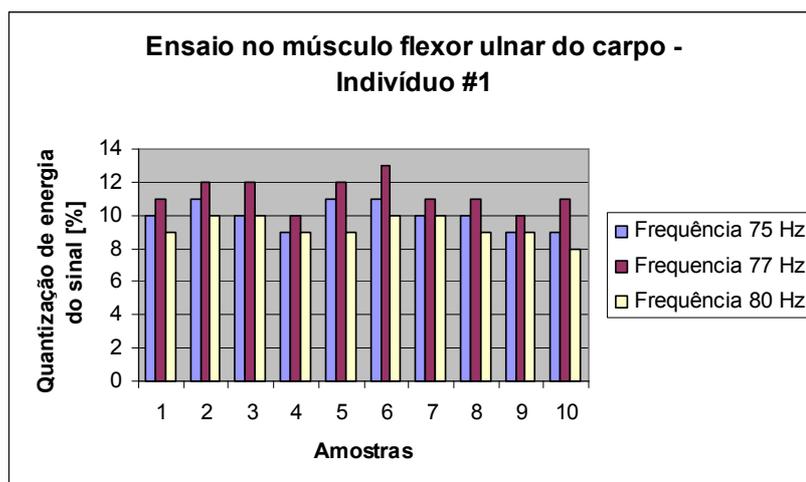


Figura 41: Gráfico dos ensaios no músculo flexor ulnar indivíduo #1.

A tabela 5 indica os ensaios no indivíduo de 67 anos medindo-se os sinais do músculo flexor profundo dos dedos. A estrutura da tabela é idêntica a anterior, onde a segunda coluna representa os sinais na frequência de 40 Hz, a terceira coluna na frequência intermediária de 42 Hz e a quarta coluna na frequência de 45 Hz.

Tabela 5: Ensaio de medição da flexão profunda dos dedos no indivíduo #2

Número do ensaio	Quantização da energia do sinal na frequência de 40 Hz[%]	Quantização da energia do sinal na frequência de 42 Hz[%]	Quantização da energia do sinal na frequência de 45 Hz[%]
1	13	15	12
2	12	14	11
3	13	16	10
4	11	16	10
5	11	15	11
6	12	14	10
7	12	14	11
8	13	15	11
9	12	15	12
10	11	14	12

O gráfico da figura 42 demonstra o comportamento do sinal nas 3 frequências avaliadas.

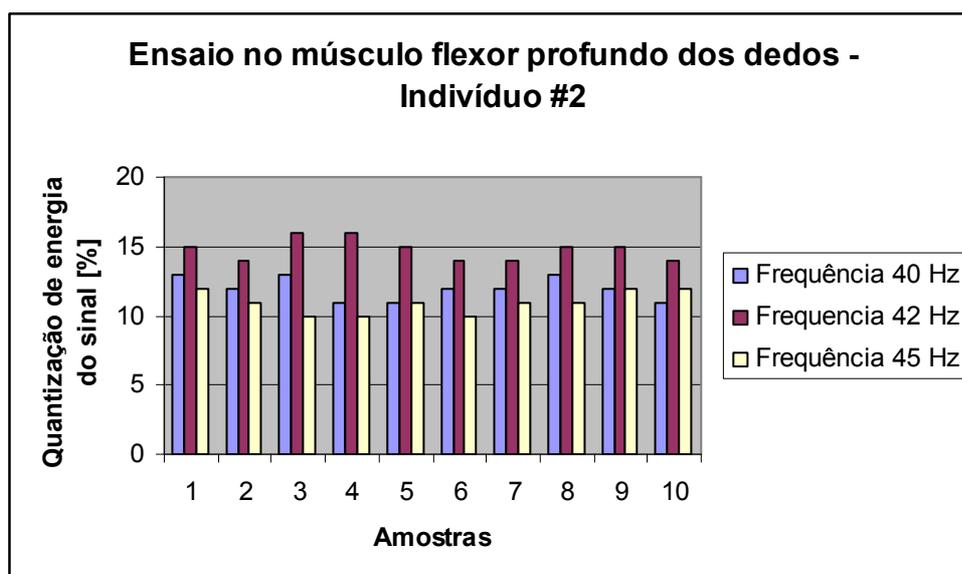


Figura 42: Gráfico dos ensaios no músculo flexor profundo dos dedos no indivíduo #2.

A tabela 6 indica os ensaios no mesmo indivíduo de 67 anos agora, medindo-se os sinais do músculo flexor ulnar do carpo. A estrutura da tabela é idêntica a anterior, onde a segunda coluna representa os sinais na frequência de 75 Hz, a terceira coluna na frequência intermediária de 77 Hz e a quarta coluna na frequência de 80 Hz.

Tabela 6: Ensaio de medição da flexão ulnar do carpo no indivíduo #2

Número do ensaio	Quantização da energia do sinal na frequência de 75 Hz[%]	Quantização da energia do sinal na frequência de 77 Hz[%]	Quantização da energia do sinal na frequência de 80 Hz[%]
1	5	7	6
2	4	7	6
3	4	8	5
4	4	8	6
5	5	7	6
6	5	8	6
7	5	8	5
8	4	8	5
9	6	7	5
10	6	7	5

O gráfico da figura 43 demonstra o comportamento do sinal nas 3 frequências avaliadas.

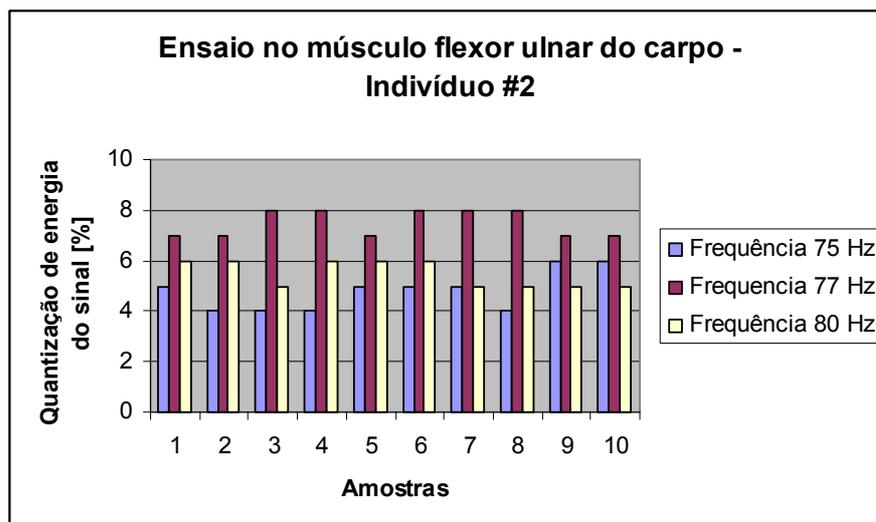


Figura 43: Gráfico dos ensaios no músculo flexor ulnar do carpo no indivíduo #2.

A tabela 7 indica os ensaios no indivíduo de 20 anos agora, medindo-se o sinal do músculo flexor profundo dos dedos. A estrutura da tabela é idêntica a anterior, onde a segunda coluna representa os sinais na frequência de 40 Hz, a terceira coluna na frequência intermediária de 42 Hz e a quarta coluna na frequência de 45 Hz.

Tabela 7: Ensaio de medição da flexão profunda dos dedos no indivíduo #3

Número do ensaio	Quantização da energia do sinal na frequência de 40 Hz[%]	Quantização da energia do sinal na frequência de 42 Hz[%]	Quantização da energia do sinal na frequência de 45 Hz[%]
1	25	29	26
2	23	28	25
3	23	28	26
4	25	30	26
5	24	29	26
6	24	28	25
7	23	29	26
8	25	30	26
9	24	30	24
10	24	31	24

O gráfico da figura 44 demonstra o comportamento do sinal nas 3 frequências avaliadas.

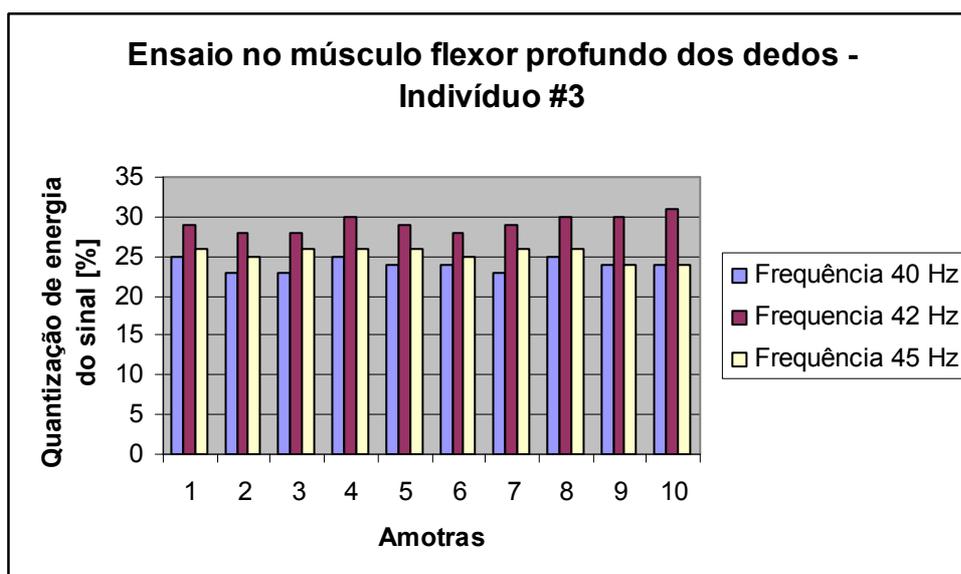


Figura 44: Gráfico dos ensaios no músculo flexor profundo dos dedos no indivíduo #3.

A tabela 8 indica os ensaios no mesmo indivíduo de 20 anos agora, medindo-se os sinais do músculo flexor ulnar do carpo. A estrutura da tabela é idêntica a anterior, onde a segunda coluna representa os sinais na frequência de 75 Hz, a terceira coluna na frequência intermediária de 77 Hz e a quarta coluna na frequência de 80 Hz.

Tabela 8: Ensaio de medição da flexão ulnar do carpo no indivíduo #3

Número do ensaio	Quantização da energia do sinal na frequência de 75 Hz[%]	Quantização da energia do sinal na frequência de 77 Hz[%]	Quantização da energia do sinal na frequência de 45 Hz[%]
1	12	16	13
2	14	17	13
3	14	16	13
4	14	16	12
5	13	15	12
6	13	16	14
7	13	16	12
8	12	15	12
9	12	15	13
10	12	15	13

O gráfico da figura 45 demonstra o comportamento do sinal nas 3 frequências avaliadas.

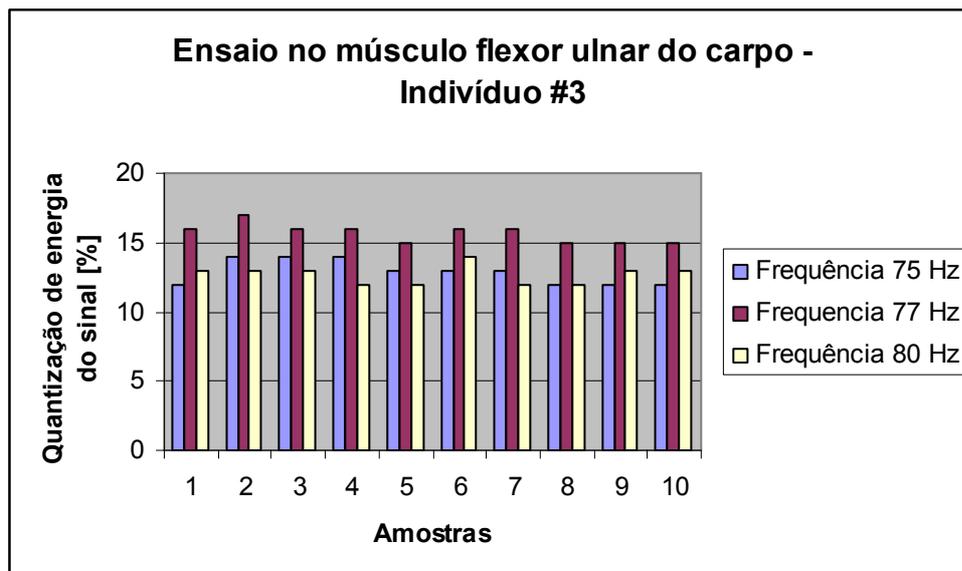


Figura 45: Gráfico dos ensaios no músculo flexor ulnar do carpo no indivíduo #3.

Aplicando um cálculo de desvio padrão experimental nos resultados dos ensaios chega-se a informações da tabela 9. Nela é possível verificar o desvio padrão em cada um dos indivíduos para as leituras feitas no músculo flexor profundo dos dedos.

Tabela 9: Desvio padrão para os sinais do músculo flexor profundo dos dedos

Número do indivíduo	Desvio padrão na frequência de 40 Hz	Desvio padrão na frequência de 42 Hz	Desvio padrão na frequência de 45 Hz
1	0,6	0,7	0,5
2	0,8	0,8	0,8
3	0,8	1,0	0,8

Na prática estes desvios representam um valor aproximado de 1 unidade dos sinais lidos. Este valor representa cerca de 6% dos valores médios para o indivíduo #1, cerca de 8% para os valores do indivíduo #2 e cerca de 4% dos valores do indivíduo #3. Realizando uma média do que representa este desvio nos três indivíduos, em valores percentuais, chega-se a um valor de aproximadamente 6 %.

Da análise dos cálculos realizados pode-se concluir que é necessário implementar melhoria na captação dos sinais, principalmente no que se refere ao um melhor posicionamento dos eletrodos a fim de se conseguir um sinal com maior intensidade e menos suscetível a interferências. Também é importante implementar rotinas de testes que efetuem uma série maior de ensaios para um mesmo indivíduo.

Tabela 10: Desvio padrão para os sinais do músculo flexor ulnar do carpo

Número do indivíduo	Desvio padrão na frequência de 75 Hz	Desvio padrão na frequência de 77 Hz	Desvio padrão na frequência de 80 Hz
1	0,8	1,0	0,7
2	0,8	0,5	0,5
3	0,9	0,7	0,7

Na prática estes desvios representam um valor aproximado de 1 unidade dos sinais lidos. Este valor representa cerca de 10% dos valores médios para o indivíduo #1, cerca de 17% para dos valores do indivíduo #2 e cerca de 7% dos valores do indivíduo #3. Realizando uma média do que representa este desvio nos três indivíduos, em valores percentuais, chega-se a um valor de aproximadamente 11 %. Pode-se concluir que, em comparação com os resultados obtidos nas leituras do músculo flexor profundo dos dedos, este resultado indica

uma incerteza maior nas leituras. Também é importante avaliar com melhor cuidado os pontos para medição deste músculo específico e melhorar a fixação dos eletrodos.

Dos ensaios nos dois tipos de músculo pode-se concluir da necessidade de um maior cuidado na leitura de sinais com menor intensidade e da necessidade de implantar proteções cada vez mais eficazes no sentido de bloquear a ação de interferências eletromagnéticas. Também verificou-se uma concentração significativa do sinal na frequência de 60 Hz que se origina principalmente dos circuito próximos onde se efetuava os testes no protótipo (lâmpadas fluorescentes, equipamentos de telecomunicação, informática, etc). Eliminar esta interferência é de vital importância para a melhoria do sinal a ser adquirido.

4.3 TESTES NO SISTEMA INTERLIGADO

Os testes com o sistema interligado têm o objetivo de validar o funcionamento geral do protótipo, ou seja, avaliar se o mesmo consegue captar os sinais dos músculos, interpretá-los e acionar a etapa de potência e por sua vez os motores responsáveis pelos movimentos.

O procedimento para este teste consiste em fixar os eletrodos na região da pele sobre o músculo flexor profundo dos dedos, realizar o movimento de fechamento da mão e verificar se houve o acionamento do motor que executa o movimento radial. Quando o movimento cessar, o motor deve retornar para o ponto de partida. Um modelo do teste pode ser visto nas figuras 46.A e 46.B. Ambas apresentam a vista superior do protótipo. Na figura 46.A a haste se encontra na posição inicial quando não há detecção de sinal da atividade muscular. Na figura 46.B a haste se encontra na posição final do movimento indicando que o movimento do músculo foi detectado. O ângulo do movimento da haste é de aproximadamente 50°. No ponto final do movimento o sistema deve manter a haste nesta posição enquanto for detectada a presença do sinal muscular. A tabela 11 mostra os resultados de 10 ensaios realizados. Nestes ensaios o usuário mantém flexionado o músculo e é verificado se o dispositivo permanece com a haste posicionada no ponto final do curso de ação. Nesta tabela a primeira coluna indica o número do ensaio, a segunda coluna o tempo em que foi mantido o músculo flexionado e na terceira coluna o tempo em que o protótipo manteve a haste na posição final.

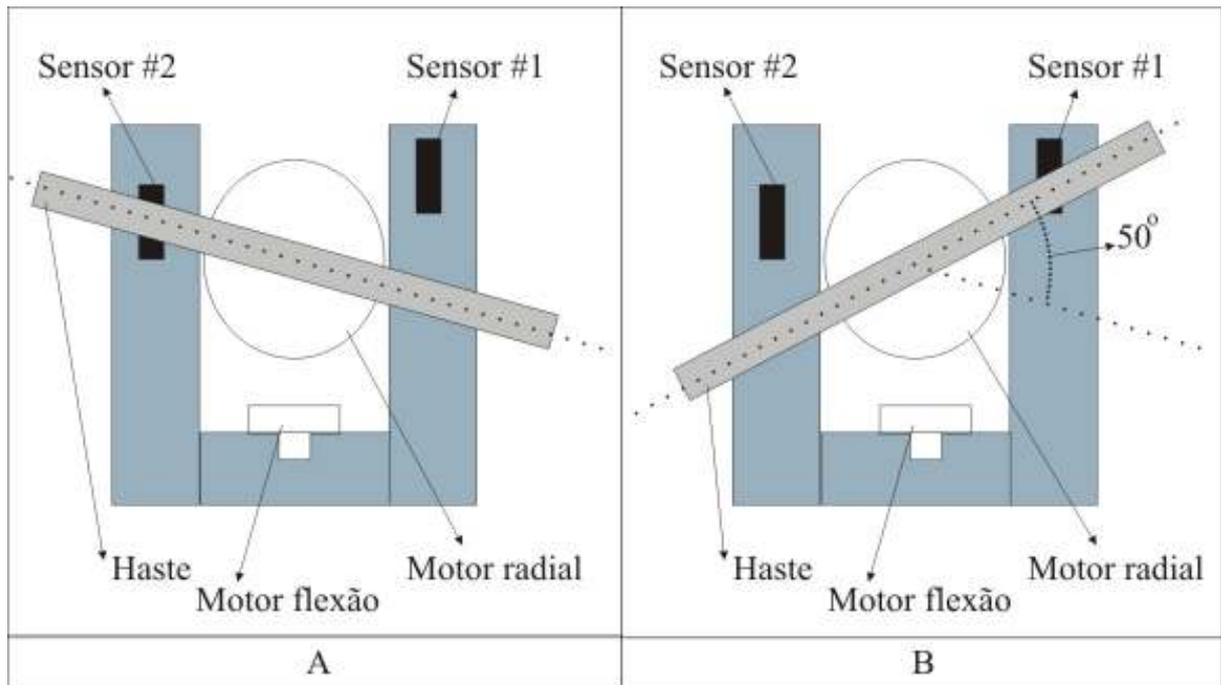


Figura 46.A: Haste na posição inicial. 46.B: Haste na posição final.

Tabela 11: Resultados do teste de funcionamento do protótipo para um tempo de 10 segundos

Número do ensaio	Tempo de atividade do músculo flexor profundo dos dedos [s]	Tempo em que a haste foi mantida na posição final [s]
1	10	8
2	10	7
3	10	8
4	10	8
5	10	8
6	10	8
7	10	8
8	10	8
9	10	7
10	10	7

Processando os dados da tabela é possível verificar que o erro máximo percentual encontrado entre a segunda e terceira coluna é de 30%.

Um segundo ensaio foi realizado considerando um tempo de 5 segundos em que se mantinha o músculo flexionado. Os resultados estão na tabela 12.

Tabela 12: Resultados do teste de funcionamento do protótipo para um tempo de 5 segundos

Número do ensaio	Tempo de atividade do músculo flexor profundo dos dedos [s]	Tempo em que a haste foi mantida na posição final [s]
1	5	4
2	5	4
3	5	5
4	5	5
5	5	5
6	5	5
7	5	5
8	5	5
9	5	4
10	5	4

Observa-se neste ensaio que o erro máximo é de 10%. Representando 1/3 do erro obtido no ensaio anterior.

Com relação aos ensaios de teste do protótipo é possível concluir que são necessárias melhorias na análise do sinal mioelétrico em períodos de maior atividade muscular. Verifica-se também que, em intervalos de tempo relativamente menores, a protótipo se comporta de maneira mais adequada, ou seja, mantém a prótese em um tempo mais próximo em relação ao tempo de atividade do músculo.

Estes mesmos ensaios foram realizados com o músculo flexor ulnar do carpo, no entanto foi possível apenas efetuar a aquisição dos sinais mioelétricos não sendo possível atuar na prótese. Em princípio verificou-se que as rotinas de análise do sinal mioelétrico, para este músculo, não mostraram resultados práticos da possibilidade de sua interpretação. Necessitando uma análise posterior com maior rigor de detalhes.

5 CONCLUSÃO

Neste trabalho de conclusão foram realizadas medições de sinais elétricos musculares (mioelétricos) dos músculos flexor profundo dos dedos e do flexor ulnar do carpo. Para isso utilizou-se um sistema microcontrolado executando um aplicativo que efetua as digitalizações dos sinais e as envia para uma unidade central. Esta unidade central, formada por um computador, efetua as interpretações dos sinais através de rotinas de processamento digital e identifica características peculiares dos mesmos. Esta unidade também atua, através de uma interface de potência, em dois motores que representam os movimentos detectados.

Para a validação do protótipo foram realizados 3 tipos de testes. Testes funcionais das partes integrantes do protótipo em separado, testes nas leituras dos sinais mioelétricos de 3 indivíduos e testes do protótipo em funcionamento com todo os seus componentes interligados.

Os testes realizados nas partes separadamente possibilitaram a validação de todos os componentes. A parte de condicionamento da interface de aquisição apresentou os ganhos desejados mediante as configurações dos amplificadores operacionais. Pequenas diferenças nos valores desejados e os obtidos se devem as tolerâncias nos valores nominais dos componentes envolvidos. Diferenças da ordem de 0,5% podem ser desconsideradas nos resultados diante da faixa nominal em que o circuito foi projetado. A parte de digitalização dos sinais na interface de aquisição apresentou um comportamento dentro do que foi estipulado, apresentando um erro de quantização na ordem de 0,4 %. Este erro, na faixa total do conversor AD representa o mesmo valor da sua resolução que é de 0,02V. Os testes da unidade central, principalmente da rotina de FFT, apresentaram um erro de 1 Hz ao longo da faixa. O que representou uma faixa de erro de 1,3 a 2,5 % no intervalo de frequências das atividades musculares estudadas. Os testes funcionais na etapa de potência e na prótese mecânica comprovaram o funcionamento dos mesmos garantindo reprodutibilidade dos movimentos e acionamentos. Cabe ressaltar a necessidade de melhorias quanto a tempos de acionamento e velocidade de resposta.

Os ensaios realizados nas aquisições dos sinais mioelétricos dos 3 indivíduos indicaram, qualitativamente, a possibilidade de identificar as frequências predominantes dos movimentos na totalidade das amostras obtidas. Em termos quantitativos, os resultados apresentam desvios médios de 6% para os sinais obtidos no músculo flexor profundo dos dedos e 11% para o músculo flexor ulnar do carpo. Estes desvios indicam a necessidade de

aprimoramentos nos processos de aquisição principalmente com relação aos sinais do músculo flexor ulnar do carpo. As principais melhorias detectadas estão em um posicionamento mais correto dos eletrodos, o uso de conversores AD com maior resolução e principalmente investimento em proteção dos dispositivos em relação a interferências eletromagnéticas.

Quando colocado em funcionamento o conjunto interligado, ou seja, o protótipo efetuando a leitura dos sinais e atuando nos motores, foi possível garantir reprodutibilidade para tempos de acionamento iguais ou menores a 5 segundos, com erros na ordem de 10%. Já para tempos superiores a 5 segundos o erro apresentado nos testes foi cerca de 30%, para o músculo flexor profundo dos dedos. Indicando a necessidade da implementação de melhorias no processo. Já os testes do protótipo para o músculo flexor ulnar do carpo não obtiveram o êxito desejado. Para este músculo foi possível efetuar aquisição do sinal, mas não se pode atuar na prótese. As causas ainda necessitam uma melhor investigação, no entanto reforça-se a necessidade de um investimento maior de tempo e recursos para implementação de melhorias.

Com relação às metas estipuladas no início do projeto, avalia-se como positivo o saldo de todo o trabalho. Foi possível efetuar a aquisição e interpretação dos sinais mioelétricos e atuar parcialmente na prótese desenvolvida.

Em face da pesquisa e dos experimentos realizados foi possível alcançar conhecimento na área de instrumentação biomédica em conjunto com os conhecimentos básicos ministrados ao longo do curso de Engenharia Elétrica. Conhecimentos que possibilitaram a manipulação das informações de maneira mais confiável possível e na obtenção de conclusões que fundamentam tais informes.

Como todo o processo tecnológico necessita de constante aprimoramento do projeto em questão também se propõem a realizar uma pesquisa constante em novas técnicas e metodologias de abordagem que venham a somar no desenvolvimento de um produto e ou serviço de melhor qualidade.

REFERÊNCIAS

BASMAJIAN, John V. **Muscles Alive**: their functions revealed by electromyography. Baltimore: Williams & Wilkins, 1962.

BEAR, Mark F.; CONNORS, Barry W.; PARADISO, Michael A. **Neurociências**: Desvendando o Sistema Nervoso. Porto Alegre: ARTMED, 2002.

DÂNGELO, José Geraldo. **Anatomia Básica dos Sistemas Orgânicos**. São Paulo: Atheneu, 2002.

DE LUCA, Carlos J. **Surface Electromyography**: Detection And Recording. Delsys, 2002. Disponível em: <http://www.delsys.com/Attachments_pdf/WP_SEMGintro.pdf>. Acesso em: 03 jun. 2008, 21:00:00.

GUYTON, Arthur C. **Tratado de Fisiologia Médica**. Rio de Janeiro: Guanabara Koogan, 1992.

_____. **Neurociência Básica**. Rio de Janeiro: Guanabara Koogan, 1993.

HAYES, Monson H. **Digital Signal Processing**. Mc Graw Hill, 1999.

IFEACHOR, E. C.; JERVIS, B. W. **Digital Signal Processing**: Pratical Approach. Addison-Weslwy, 1993.

LEPPLE, Charles. **The Fast Fourier Transform Demystified**. Thomas Jefferson High School for Science and Technology Microelectronics Tech, 1996.

LUECKE, Gerald. **Analog and Digital Circuits for Electronic Control System**: using the TI MSP430 microcontroller. Elsevier: Burlington, 2005.

MACHADO, Angelo B. M. **Neuroanatomia Funcional**. São Paulo: Atheneu, 2006.

MIRANDA, Edalton. **Bases de Anatomia e Cinesiologia**. Rio de Janeiro: Sprint, 2006.

OPPENHEIM, Alan V.; SCHAFER, Ronald W.; BUCK, John R. **Discrete-Time Signal Processing**. 2nd ed. New Jersey: Prentice Hall, 1998.

Pennsylvania State University. **Athletic Training Research Laboratory**, 2008. Disponível em: <http://www.hhdev.psu.edu/atlab/photo_gallery.html>. Acesso em: 03 jun. 2008, 21:30:00.

PEREZ, Vincent. **Atlas do Corpo Humano**. São Paulo: Barros, Fischer & Associados, 2005.

SPIEGEL, Murray Ralph. **Análise de Fourier** / Murray Ralph Spiegel ; tradução [de] Alfredo Alves de Faria. São Paulo: McGraw-Hill, 1977.

Universidade Federal de Pernambuco. **Laboratório de Dispositivos e Nanoestruturas**. Recife, 2008. Disponível em:<<http://www2.ee.ufpe.br/codec/DTF%20E%20FFT.pdf>>. Acesso em: 06 jun. 2008, 10:45:00.

VILELA, Ana Luisa Miranda. **Sistema Muscular**. 2008. Disponível em: <<http://www.afh.bio.br/sustenta/Sustenta4.asp>>. Acesso em: 01 abr. 2008, 20:30:10.

WEBSTER, John G. **Encyclopedia of Medical Devices and Instrumentation**. 2nd. ed. New Jersey: John Wiley & Sons, 2006.

OBRAS CONSULTADAS

AHMED, Ashfaq. **Eletrônica de Potência**. São Paulo: Pearson Prentice Hall, 2000.

BACK, Nelson. **Metodologia de Projeto de Produtos Industriais**. Rio de Janeiro: Guanabara Koogan, 1983.

GONÇALVES, Hortência de Abreu. **Manual de Monografia, Dissertação e Tese**. São Paulo: Avercamp, 2004.

LIMA, Teófilo Lourenço de. **Manual Básico para Elaboração de Monografia**. Canoas: ULBRA, 1999.

LUENBERGER, David G. **Linear and Nonlinear Programming**. New York: Springer, 2003.

MONTGOMERY, Douglas C.; RUNGER, George C.; HEBELE, Norma F. **Estatística Aplicada à Engenharia**. 2. ed. Rio de Janeiro: LTC, 2004.

PAHL, Gerhard; BEITZ Wolfgang; FELDHUSEN, Jorg; GROTE, Karl-Heinrich. **Projeto na Engenharia**. Berlin: Edgard Blucher, 2005.

ROSS, Sheldon M. **Introduction to Probability Models**. Burlington: Elsevier, 2007.

SANTOS, João Carlos Verneti dos. **Introdução à Teoria de Redes**. Canoas: Editora da ULBRA, 2003.

TOMASZEWSKI, Jairo Rodrigo. **TCC – Estudo e Desenvolvimento de um Sistema Experimental para Controle de Prótese Parcial de Mão**. Canoas: 2005.

APÊNDICES

APÊNDICE A – FONTE DO PROGRAMA DA INTERFACE DE POTÊNCIA

/*

TRABALHO DE CONCLUSAO DE CURSO

ENGENHARIA ELETRICA

POR : Fabio de Araujo Bairros

NOME : INTERFACE PARA ACIONAMENTO DE MOTOR DE PASSO PARA CONTROLE DE

BRACO MECANICO

DISCIPLINA : TCC I E TCC II

CLIENTE : ULBRA

DATA : 04/02/2008

*/

```
#include <16f72.h> //Inclui cabeçalho do componente

#device adc=8;//DEFINE COMO 8 BITS PARA O AD

#fuses HS, NOWDT, PUT, NOBROWNOUT, NOPROTECT, XT //Define estado dos fusíveis

//NOLVP

#use delay(clock=4000000) //Define o valor do clock

#include <rs232.c> //rotinas de que usaram o canal serial//RS-232

//#use fixed_io (a_outputs = pin_a0, pin_a1, pin_a2, pin_a3, pin_a4, pin_a5)
//Define porto A como saída

#use fixed_io (b_outputs = pin_b0, pin_b1, pin_b2, pin_b3, pin_b4, pin_b5, pin_b6, pin_b7) //Define o
Porto B como saída

//#use fixed_io (c_outputs = pin_c0, pin_c1, pin_c3, pin_c5)

int contador;

int valor_adc; //RECEBE O VALOR DA CONVERÇÃO DO AD

int i_dado_recebido; //DADOS RECEBIDO PELA SERIAL

int tempo_motor; //TEMPO DE DELAY PARA O PASSO DO MOTOR

int valor_teclado; //RECEBE O VALOR DO TECLADO

//0 -> nenhuma tecla

//1 -> tecla direita

//2 -> tecla esquerda
```

```

int          valor_motor;          //GUARDA O VALOR ATUAL O MOTOR

boolean     status_sentido;        //INDICA EM QUE SENTIDO O MOTOR ESTA GIRANTO
NESTE MOMENTO

                                                //0 -> HORARIO
                                                //1 -> ANTI-HORARIO

boolean     leitura_sensor;        //RECEBE O VALOR DE UM DETERMINADO PINO

boolean     sensor_direita;        //GUARDA ESTADO DO SENSOR DA DIREITA - ATIVADO
PELO MOVIMENTO NO SENTIDO HORARIO

boolean     sensor_esquerda;      //GUARDA ESTADO DO SENSOR DA ESQUERDA - ATIVADO PELO
MOVIMENTO NO SENTIDO ANTI-HORARIO

boolean     tecla_direita;         //GUARDA O VALOR DA TECLA LIGADA AO PINO C2 ->
TECLA DIREITA

boolean     tecla_esquerda;       //GUARDA O VALOR DA TECLA LIGADA AO PINO C3 ->
TECLA ESQUERDA

boolean     tecla_1;              //TECLAD DA DIREITA

boolean     tecla_2;              //TECLA DA ESQUERDA

//#####/
//LEITURA DO ADC - CANAL ZERO - ONDE SE ENCONTRA O SINAL DA DO SENSOR DE CORRENTE

void le_adc_0(void)

{

#asm

InitializeAD_2:

                bsf 0x03, 5 ; //select Bank1

                movlw 0x00 ; //select RA3-RA0

                movwf 0x9f ; //as analog inputs - ADCON1

                bcf 0x03, 5 ; //select Bank0

                movlw 0x41 ; //select: OSC/8, -> CANAL 0 -> 01 000 001

                movwf 0x1f ; //turn on A/D - ADCON0

//Convert call sample-delay ; provide necessary sampling time

                bsf 0x1f, 2 ; //start new A/D conversion - ADCON0 - bit 2

loop_2:

                btfsc 0x1f, 2 ; //A/D over? - ASCON0 - bit2

```

```

        goto loop_2 ; //no then loop

        movf 0x1e, w ; //yes then get A/D value - ADRES

        movwf valor_adc;

#endasm
}

//#####/
/DESLIGAR MOTOR PUNHO -> COLOCA TODOS OS PINO EM ZERO

void para_motor_punho(void)
{
    output_low(pin_b0);
    output_low(pin_b1);
    output_low(pin_b2);
    output_low(pin_b3);
    delay_ms(10);
}

//#####/
/DESLIGAR MOTOR MAO -> COLOCA TODOS OS PINO EM ZERO

void para_motor_mao(void)
{
    output_low(pin_a0);
    output_low(pin_a1);
    output_low(pin_a2);
    output_low(pin_a3);
    delay_ms(10);
}

//#####/
/LEITURA DOS SENSORES

void le_sensores_punho(void)
{
    sensor_direita = input(pin_c1);
    sensor_esquerda = input(pin_c0);
}

```

```

}

//#####/
/LEITURA DOS SENSORES

void le_sensores_mao(void)

{
    sensor_direita = input(pin_c1);
    sensor_esquerda = input(pin_c0);
}

//#####/
/ACIONAMENTO DO MOTOR QUE EFETUA O GIRO DO PUNHO -> SENTIDO HORARIO

void motor_punho_horario(void)

{
    valor_motor = 8;                //Inicia com valor 8
    para_motor_punho();            //Zera as saida das bobinas
    le_sensores_punho();           //Le os sensores antes de entrar no laco
    while(!sensor_esquerda)
    {
        output_b(valor_motor);
        delay_ms(tempo_motor);
        para_motor_punho();
        valor_motor = valor_motor >> 1;    //Desloca o valor para direita uma vez
        if(valor_motor <= 0)
        {
            valor_motor = 8;
        }
        le_sensores_punho();
    }
    para_motor_punho();
}

//#####/
/ACIONAMENTO DO MOTOR QUE EFETUA O GIRO DO PUNHO -> SENTIDO ANTI-HORARIO

```

```

void motor_punho_anti_horario(void)
{
    valor_motor = 1;                //Inicia com valor 8
    para_motor_punho();             //Zera as saida das bobinas
    le_sensores_punho();            //Le os sensores antes de entrar no laco
    while(!sensor_direita)
    {
        output_b(valor_motor);
        delay_ms(tempo_motor);
        para_motor_punho();
        valor_motor = valor_motor << 1;    //Desloca o valor para esquerda uma vez
        if(valor_motor >= 16)
        {
            valor_motor = 1;
        }
        le_sensores_punho();
    }
    para_motor_punho();
}

#####
/ACIONAMENTO DO MOTOR QUE EFETUA O FECHAMENTO DA MAO -> SENTIDO HORARIO
void motor_mao_horario(void)
{
    valor_motor = 8;                //Inicia com valor 8
    para_motor_mao();               //Zera as saida das bobinas
    le_sensores_mao();              //Le os sensores antes de entrar no laco
    while(!sensor_esquerda)
    {
        output_a(valor_motor);        //No porto A este localizado os sinais do motor
    }
}

```

```

        delay_ms(tempo_motor);

        para_motor_mao();

        valor_motor = valor_motor >> 1;           //Desloca o valor para direita uma vez

        if(valor_motor <= 0)

        {

                valor_motor = 8;

        }

        le_sensores_mao();

    }

    para_motor_mao();

}

//#####/
/ACIONAMENTO DO MOTOR QUE EFETUA O FECHAMENTO DA MAO -> SENTIDO ANTI-HORARIO

void motor_mao_anti_horario(void)

{

        valor_motor = 1;                           //Inicia com valor 8

        para_motor_mao();                           //Zera as saida das bobinas

        le_sensores_mao();                           //Le os sensores antes de entrar no laco

        while(!sensor_direita)

        {

                output_a(valor_motor);               //No porto A esta localizado os sinais do motor

                delay_ms(tempo_motor);

                para_motor_mao();

                valor_motor = valor_motor << 1;       //Desloca o valor para esquerda uma vez

                if(valor_motor >= 16)

                {

                        valor_motor = 1;

                }

                le_sensores_mao();

        }

}

```

```

        para_motor_mao();
    }

    //#####

    *

FUNCAO PRINCIPAL - CORPO DO PROGRAMA

void main(void)
{
    set_tris_c(0xff);           //Todos os pinos do porto C como entrada
    rs232_inicializa();        //Inicializa canal serial com parametros de velocidade, controle
    para_motor_mao();          //Chama funcao para desligar motor
    para_motor_punho();        //Chama funcao para desligar motor
    tempo_motor = 100;
    output_low(pin_b7);
    motor_punho_horario();
    while(1)
    {
        i_dado_recebido = rs232_recebe();           //Aguarda o dado pela serial
        switch(i_dado_recebido)
        {
            case 'a':
                output_high(pin_b7);
                motor_punho_anti_horario();
                break;

            case 'd':
                output_low(pin_b7);
                motor_punho_horario();
                break;

            case '3':
                output_high(pin_b7);
                motor_mao_horario();
        }
    }
}
//

```

```
break;
case '4':
    output_low(pin_b7);
    motor_mao_anti_horario();
    break;
default :
    break;
}
}
}
```

APÊNDICE B – FONTE DO PROGRAMA DA INTERFACE DE AQUISIÇÃO

TRABALHO DE CONCLUSÃO DE CURSO

ENGENHARIA ELETRICA

POR : Fabio de Araujo Bairos

NOME : INTERFACE PARA ACIONAMENTO DE MOTOR DE PASSO PARA CONTROLE DE
BRACO MECANICO

DISCIPLINA : TCC I E TCC II

CLIENTE : ULBRA

DATA : 21/09/2007

*/

```
#include <16f72.h> //Inclui cabeçalho do componente

#define adc=8; //DEFINE COMO 8 BITS PARA O AD

#define HS, NOWDT, PUT, NOBROWNOUT, NOPROTECT, XT //Define estado dos fusíveis

//NOLVP

#define delay(clock=4000000) //Define o valor do clock

//#use rs232(baud=9600, xmit=PIN_C7, rcv=PIN_C6) //Estabelece que ira utilizar

#include <rs232.c> //rotinas de que usaram o canal serial RS-232

//#use fixed_io (a_outputs = pin_a0, pin_a1, pin_a2, pin_a3, pin_a4, pin_a5)
//Define porto A como saída

#define fixed_io (b_outputs = pin_b0, pin_b1, pin_b2, pin_b3, pin_b4, pin_b5, pin_b6, pin_b7) //Define o
Porto B como saída

#define fixed_io (c_outputs = pin_c0, pin_c1, pin_c3, pin_c5)

int contador;

int valor_adc; //RECEBE O VALOR DA CONVERÇÃO DO AD

#####
/LEITURA DO ADC - CANAL ZERO - ONDE SE ENCONTRA O SINAL DA DO SENSOR DE CORRENTE

void le_adc_0(void)

{

#asm

InitializeAD_2:
```

```

        bsf 0x03, 5 ; //select Bank1

        movlw 0x00 ; //select RA3-RA0

        movwf 0x9f ; //as analog inputs - ADCON1

        bcf 0x03, 5 ; //select Bank0

        movlw 0x41 ; //select: OSC/8, -> CANAL 0 -> 01 000 001

        movwf 0x1f ; //turn on A/D - ADCON0

//Convert call sample-delay ; provide necessary sampling time

        bsf 0x1f, 2 ; //start new A/D conversion - ADCON0 - bit 2

loop_2:

        btfsc 0x1f, 2 ; //A/D over? - ASCON0 - bit2

        goto loop_2 ; //no then loop

        movf 0x1e, w ; //yes then get A/D value - ADRES

        movwf valor_adc;

#endasm
}

//#####

*

FUNCAO PRINCIPAL - CORPO DO PROGRAMA

void main(void)

{

    set_tris_c(0xff);                //Todos os pinos do porto C como entrada

    setup_adc(ADC_CLOCK_DIV_2);

    //Define o tipo de clock a ser utilizado - clock / 2

    setup_adc_ports(ALL_ANALOG);     //Define quais serão as entradas analogicas - todas - PORTA

    rs232_inicializa();              //Inicializa canal serial com parametros de velocidade, controle

    while(1)

    {

        le_adc_0();

        rs232_transmite(valor_adc); //

    }

}

```

APÊNDICE C – FONTE DO PROGRAMA DA UNIDADE CENTRAL

```

/*
TCC -> CURSOS DE ENG. ELETRICA -> ULBRA - CANOAS/RS
DATA : 09/03/2008
AUTOR : Fabio de Araujo Bairros
DESCRICA0 :
ESTE PROGRAMA EFETUA A AQUISICAO DE DADOS DE UM SISTEMA
QUE MEDE SINAIS MIOELETRICOS E PROCESSA-OS UTILIZANDO
FERRAMENTAS DE PROCESSAMENTO DIGITAL DE SINAIS PARA
IDENTIFICAR CARACTERISTICAS DO SINAIS

#include <vcl.h>
#include <math.h>
#pragma hdrstop
#include "fft.h"
#include "Unit_config_porta.h"
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm_principal *Form_principal;
typedef struct // estrutura que guarda números complexos
{
    float real, imaginario;
}
NUMERO_COMPLEXO;
void funcao_fft(NUMERO_COMPLEXO *x, int m); // protótipo da função fft
float f_buffer_sinal[1000]; // Recebe os valores do sinal original
float f_frequencia; // Valores de frequencia
float f_magnitude_fft; // Recebe o valor da amplitude do sinal
float f_valor_inferior; // Guarda o valor da borda inferior da amplitude do sinal lido do musculo
TCommThread *Serial; // Define uma variavel ponteiro do tipo Thread // para receber as rotinas de acesso a serial
TStringList *lista_dispositivos; // Define uma variavel tipo TListStrin

```

```

// para servir de intermediario entre os tipos de dispositivos informados pela Thread Serial e o ComboBox_porta
// do Form_configuracao

int  velocidade;

int  tamanho_vetor;

int  i_ganho_sinal;//Determina um fator de multiplicação para o sinal de entrada
int  i_ganho_fft;//Fator de multiplicação para amstragem do calculo fft no gráfico
int  i_status_mao;//Indica o status para acionamento da mao enviando um sinal pela serial//até a célula
correspondente para controle

        //0 -> mao aberta
        //1 -> mao fechada

int  i_tipo_janelamento;//Determina qual o tipo de janelamento sera impregado

        //0 -> nenhum
        //1 -> Hanning
        //2 -> Hamming
        //3 -> Kaiser-Bessel
        //4 -> Triangular
        //5 -> Flattop
        //6 -> Exponencial

long int  i_auxiliar;

unsigned char  uc_auxiliar;

AnsiString porta;

AnsiString dados_sinal;

AnsiString s_estado_serial_1;//Guarda o te

__fastcall TForm_principal::TForm_principal(TComponent* Owner)
        : TForm(Owner)
{
}

//-----
void __fastcall TForm_principal::FormCreate(TObject *Sender)
{

```

```

Serial = new TCommThread();//Cria uma instancia da serial

Serial->ReturnMethod = ReturnMethod;//Define o metodo para recebimento de dados pela serial

lista_dispositivos = new TStringList;

Serial->GetAvailableDevicesNames(1,0, lista_dispositivos);

    //Utiliza o metodo da Thread para avaliar quais

    //as portas disponiveis no sistema

porta = "COM6";

velocidade = 19200;

Chart_sinal->LeftAxis->Minimum = -150;

Chart_sinal->LeftAxis->Maximum = 260;

//Eixo - x -> tempo - máximo de 1000ms -> 1hz

//incremento de 1ms

//total de 1000 incrementos

Chart_sinal->BottomAxis->Minimum = 0;

Chart_sinal->BottomAxis->Maximum = 500;

Chart_sinal->BottomAxis->Increment = 1,70;

Chart_fft->LeftAxis->Minimum = 0;

Chart_fft->LeftAxis->Maximum = 100;

Chart_fft->BottomAxis->Increment = 1,70;

Chart_fft->BottomAxis->Minimum = 0;

Chart_fft->BottomAxis->Maximum = 100;

f_frequencia = 5;

Timer2->Enabled = False;

Timer1->Enabled = True;

Button_finaliza->Enabled = False;

Button_inicia->Enabled = True;

i_ganho_sinal = 1;//Inicia com valor neutro da

    //multiplicação

i_ganho_fft = 1;//Inicia com fator neutro da multiplicação

i_status_mao = 0;//Inicializa com zero

```

```

i_tipo_janelamento = 0;

f_valor_inferior = 0.2;//Iniciliza valor com valor montado experimentalmente

try
{
    Label_valor_inferior->Caption = FormatFloat("0.00", f_valor_inferior);
}
catch(...)
{
    ;
}

s_estado_serial_1 = "Porta -> " + porta + "-" +
    IntToStr(velocidade);

Label_estado_serial_1->Caption = s_estado_serial_1;
}

void __fastcall TForm_principal::RadioButton1Click(TObject *Sender)
{
    //Onda padrão Seno

    float f_buffer[500];//Com 128 espacos para acomodar os valores da funcao

    int i_m;

    int i_largura_fft;

    i_m = 10;

    i_largura_fft = 1 << i_m;

    NUMERO_COMPLEXO *amostra;//Cria estrutura com o nome de amostra

    if(RadioButton1->Checked)
    {
        Series1->Clear();

        Series2->Clear();

        for(int contador = Chart_sinal->BottomAxis->Minimum;
            contador <= Chart_sinal->BottomAxis->Maximum;
            contador++)
        {

```

```

Series1->AddXY(contador,
    100*sin((contador / Chart_sinal->BottomAxis->Maximum) *
    2*M_PI * f_frequencia), "", clGreen);
f_buffer_sinal[contador] = 100*sin((contador /
    Chart_sinal->BottomAxis->Maximum) *
    2*M_PI * f_frequencia);
}
amostra = (NUMERO_COMPLEXO *) calloc(i_largura_fft, sizeof(NUMERO_COMPLEXO));
for( int contador = Chart_sinal->BottomAxis->Minimum;
    contador <= Chart_sinal->BottomAxis->Maximum;
    contador++)
{
//    amostra[contador].real = f_buffer_sinal[contador];
amostra[contador].real = 4* (0.5 - 0.5 * (cos((2*M_PI*contador)/
    Chart_sinal->BottomAxis->Maximum))) *
    f_buffer_sinal[contador];
}
//    amostra[contador].real = uc_auxiliar *
//        (0.5 - 0.5 * (cos((2*M_PI*contador)/tamanho_vetor)));
funcao_fft(amostra, i_m);
for(int contador = 0; contador <= 499; contador++)
{
    Series2->AddXY(contador * .9559,
        (pow(pow(amostra[contador].real, 2) +
        pow(amostra[contador].imaginario, 2), 0.5) / 1000),
        "", clRed);
}
}
}
}
void funcao_fft(NUMERO_COMPLEXO *x, int m)// função fft

```

```

{
    static NUMERO_COMPLEXO *w;

    static int mstore = 0;

    static int n = 1;

    NUMERO_COMPLEXO u, temp, tm;

    NUMERO_COMPLEXO *xi, *xip, *xj, *wptr;

    int i, j, k, l, le, windex;

    double arg, w_real, w_imag, wrecur_real, wrecur_imag, wtemp_real;

    if(m != mstore)// libera memória alocada previamente e aloca novo m
    {
        if(mstore != 0)
        {
            free(w);
        }

        mstore = m;

        if(m == 0)
        {
            return;
        }

        n = 1 << m; // n = 2**m = tamanho da fft

        le = n/2;

        w = (NUMERO_COMPLEXO *) calloc(le-1, sizeof(NUMERO_COMPLEXO)); // aloca a memória
para o w

        if(!w)
        {
            exit(1);
        }

        arg = 4.0 * atan(1.0)/le; //calcula os valores de w recursivamente

        wrecur_real = w_real = cos(arg);

        wrecur_imag = w_imag = -sin(arg);

```

```

xj = w;
for (j=1; j<le; j++)
{
    xj->real = (float)wrecur_real;
    xj->imaginario = (float)wrecur_imag;
    xj++;
    wtemp_real = wrecur_real * w_real - wrecur_imag * w_imag;
    wrecur_imag = wrecur_real * w_imag + wrecur_imag * w_real;
    wrecur_real = wtemp_real;
}
}
le = n; // começa a fft
windex = 1;
for ( l=0; l<m; l++)
{
    le = le / 2;
    for(i=0; i<n; i = i + 2*le)
    { // primeira iteração sem multiplicações
        xi = x + i;
        xip = xi + le;
        temp.real = xi->real + xip->real;
        temp.imaginario = xi->imaginario + xip->imaginario;
        xip->real = xi->real - xip->real;
        xip->imaginario = xi->imaginario - xip->imaginario;
        *xi = temp;
    }
    wptr = w + windex - 1; // iterações restantes usam w guardado
    for(j = 1; j<le; j++)
    {
        u = *wptr;

```

```

for(i=j; i<n; i = i + 2*le)
{
    xi = x + i;
    xip = xi + le;
    temp.real = xi->real + xip->real;
    temp.imaginario = xi->imaginario + xip->imaginario;
    tm.real = xi->real - xip->real;
    tm.imaginario = xi->imaginario - xip->imaginario;
    xip->real = tm.real * u.real - tm.imaginario * u.imaginario;
    xip->imaginario = tm.real * u.imaginario + tm.imaginario * u.real;
    *xi = temp;
}
wptr = wptr + windex;
}
windex = 2 * windex;
}
j = 0; // rearranja os dados revertendo os bits
for (i=1; i < (n-1); i++)
{
    k = n/2;
    while(k <= j)
    {
        j = j - k;
        k = k/2;
    }
    j = j + k;
    if (i<j)
    {
        xi = x + i;
        xj = x + j;

```

```

        temp = *xj;

        *xj = *xi;

        *xi = temp;

    }

}

}

/*

```

Código fonte arquivo Unit1.cpp:

```

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "Unit1.h"

#pragma package(smart_init)

#pragma link "SLScope"

#pragma resource "*.dfm"

typedef struct { // estrutura que guarda números complexos

float real, imag;

}COMPLEX;

void fft(COMPLEX *x, int m); // protótipo da função fft

TForm1 *Form1;

__fastcall TForm1::TForm1(TComponent* Owner)

: TForm(Owner)

{

}

void __fastcall TForm1::Timer1Timer(TObject *Sender)

{

    float Buffer[ 512 ]; // buffer que guardará os valores do módulo do sinal

    COMPLEX *samp; // buffer que guarda os valores para a realização da fft

    int m = 10, fft_length;

    fft_length = 1 << m; // obtenção do tamanho da fft

```

```

samp = (COMPLEX *) calloc(fft_length, sizeof(COMPLEX)); // alocação de memória para o buffer samp
BOOL m_bPortReady; // variáveis necessárias para a configuração serial
HANDLE m_hComm;
DCB m_dcb;
COMMTIMEOUTS m_CommTimeouts;
BOOL bReadRC;
DWORD iBytesRead;
DWORD dwSize;
DWORD dwRead;
unsigned char RxChar[ 1024 ]; // buffer que receberá os dados da serial
m_hComm = CreateFile("Com2", // obtenção da porta serial
    GENERIC_READ,
    0,
    NULL,
    OPEN_EXISTING,
    0,
    NULL);
m_bPortReady = SetupComm(m_hComm, 1024, 1024); // tamanho do buffer da serial
m_bPortReady = GetCommState(m_hComm, &m_dcb); // configuração da porta serial
m_dcb.BaudRate = 19200;
m_dcb.ByteSize = 8;
m_dcb.Parity = NOPARITY;
m_dcb.StopBits = ONESTOPBIT;
m_dcb.fAbortOnError = FALSE;
m_bPortReady = SetCommState(m_hComm, &m_dcb);
m_bPortReady = GetCommTimeouts (m_hComm, &m_CommTimeouts); // configuração de timeouts para a
serial
m_CommTimeouts.ReadIntervalTimeout = 50;
m_CommTimeouts.ReadTotalTimeoutConstant = 50;
m_CommTimeouts.ReadTotalTimeoutMultiplier = 10;

```

```

m_CommTimeouts.WriteTotalTimeoutConstant = 0;
m_CommTimeouts.WriteTotalTimeoutMultiplier = 0;
m_bPortReady = SetCommTimeouts (m_hComm, &m_CommTimeouts);
bReadRC = ReadFile(m_hComm, &RxChar, 1024, &iBytesRead, NULL); // leitura dos dados da serial
CloseHandle(m_hComm); // fechamento da porta serial
for( int i = 0; i < 1024; i ++ )
{
    samp[ i ].real = RxChar[ i ] / 51.2; // transferencia em escala dos dados lidos
}
fft(samp, m); // chamada da função fft
for(int i = 0; i < 512; i ++)
{
    Buffer[ i ] = pow(pow(samp[ i ].real, 2) + pow(samp[ i ].imag, 2), 0.5) / 1024;
    // transferência do módulo do sinal da fft
    SLScope1->Channels->Channels[ 0 ]->Data->SetYData( Buffer, 512 ); // transferência dos dados para o
gráfico
}
void fft(COMPLEX *x, int m) { // função fft
static COMPLEX *w;
static int mstore = 0;
static int n = 1;
COMPLEX u, temp, tm;
COMPLEX *xi, *xip, *xj, *wptr;
int i, j, k, l, le, windex;
double arg, w_real, w_imag, wrecur_real, wrecur_imag, wtemp_real;
if(m != mstore) { // libera memória alocada previamente e aloca novo m
if(mstore != 0) free(w);
mstore = m;
if(m == 0) return;

```

```

n = 1 << m; // n = 2**m = tamanho da fft

le = n/2;

w = (COMPLEX *) calloc(le-1, sizeof(COMPLEX)); // aloca a memória para o w

if(!w) {

exit(1);

}

arg = 4.0 * atan(1.0)/le; //calcula os valores de w recursivamente

wrecur_real = w_real = cos(arg);

wrecur_imag = w_imag = -sin(arg);

xj = w;

for (j=1; j<le; j++) {

xj->real = (float)wrecur_real;

xj->imag = (float)wrecur_imag;

xj++;

wtemp_real = wrecur_real * w_real - wrecur_imag * w_imag;

wrecur_imag = wrecur_real * w_imag + wrecur_imag * w_real;

wrecur_real = wtemp_real;

}

}

le = n; // começa a fft

windex = 1;

for ( l=0; l<m; l++) {

le = le / 2;

for(i=0; i<n; i = i + 2*le) { // primeira iteração sem multiplicações

xi = x + i;

xip = xi + le;

temp.real = xi->real + xip->real;

temp.imag = xi->imag + xip->imag;

xip->real = xi->real - xip->real;

```

```

xip->imag = xi->imag - xip->imag;

*xi = temp;
}

wptr = w + windex - 1; // iterações restantes usam w guardado

for(j = 1; j<le; j++) {

u = *wptr;

for(i=j; i<n; i = i + 2*le) {

xi = x + i;

xip = xi + le;

temp.real = xi->real + xip->real;

temp.imag = xi->imag + xip->imag;

tm.real = xi->real - xip->real;

tm.imag = xi->imag - xip->imag;

xip->real = tm.real * u.real - tm.imag * u.imag;

xip->imag = tm.real * u.imag + tm.imag * u.real;

*xi = temp;

}

wptr = wptr + windex;

}

windex = 2 * windex;

}

j = 0; // rearranja os dados revertendo os bits

for (i=1; i < (n-1); i++) {

k = n/2;

while(k <= j) {

j = j - k;

k = k/2;

}

j = j + k;

if (i<j) {

```

```

xi = x + i;
xj = x + j;
temp = *xj;
*xj = *xi;
*xi = temp;
}
}
}
*/

void __fastcall TForm_principal::RadioButton2Click(TObject *Sender)
{
float  f_buffer[512]; //Com 128 espacos para acomodar os valores da funcao
int    i_m;
int    i_largura_fft;
        i_m = 10;
        i_largura_fft = 1 << i_m;
        NUMERO_COMPLEXO *amostra; //Cria estrutura com
                //o nome de amostra

Series1->Clear();
Series2->Clear();
if(RadioButton2->Checked)
{
        for(int contador = Chart_sinal->BottomAxis->Minimum;
                contador <= (Chart_sinal->BottomAxis->Maximum) / 2;
                contador++)
        {
                Series1->AddXY(contador, Chart_sinal->LeftAxis->Maximum,
                        "", clGreen);
                f_buffer_sinal[contador] = Chart_sinal->LeftAxis->Maximum;
        }
}

```

```

for(int contador = Chart_sinal->BottomAxis->Maximum / 2;
    contador <= Chart_sinal->BottomAxis->Maximum;
    contador++)
{
    Series1->AddXY(contador,Chart_sinal->LeftAxis->Minimum,
        "", clGreen);
    f_buffer_sinal[contador] = Chart_sinal->LeftAxis->Minimum;
}
amostra = (NUMERO_COMPLEXO *) calloc(i_largura_fft, sizeof(NUMERO_COMPLEXO));
for( int contador = Chart_sinal->BottomAxis->Minimum;
    contador <= Chart_sinal->BottomAxis->Maximum;
    contador++)
{
    amostra[contador].real = f_buffer_sinal[contador];
}
funcao_fft(amostra, i_m);
for(int contador = 0; contador <= 511; contador++)
{
    Series2->AddXY(contador,
        (pow(pow(amostra[contador].real, 2) +
            pow(amostra[contador].imaginario, 2), 0.5) / 256),
        "", clRed);
}
}
}
void __fastcall TForm_principal::RadioButon_onlineClick(TObject *Sender)
{
    Button_inicia->Enabled = False;
    Button_finaliza->Enabled = False;
}

```

```
try
{
//      porta = "COM6";
//      velocidade = 19200;

Serial->SetDeviceName(porta);//Define a porta a ser usada

Serial->Connect();//Efetua a conexao

if(Serial->Connect())//Verifica se a porta foi conectada

    //

    {

        Shape1->Brush->Color = clRed;

        Serial->SetBaudRate(velocidade);

        Serial->SetByteSize(8);

        Serial->SetParity(NOPARITY);

        Serial->SetStopBits(ONESTOPBIT);

        Panel3->Caption = "";

        Panel4->Caption = "";

        Memo1->Clear();

    }

else

    {

        Shape1->Brush->Color = clWhite;

    }

    Series1->Clear();

}

catch(...)

{

    ;

}

Timer2->Enabled = True;
```

```

}

void __fastcall TForm_principal::MaskEdit_frequenciaKeyPress(TObject *Sender,
char &Key)
{
    if(Key == 13)
    {
        try
        {
            f_frequencia = StrToFloat(MaskEdit_frequencia->Text)/2;
            Series1->Clear();
            Series2->Clear();
            RadioButton1->Checked = False;
            RadioButton2->Checked = False;
            RadioButton_online->Checked = False;
        }
        catch(...)
        {
            ;
        }
    }
}

/*
#define NZEROS 4
#define NPOLES 4
#define GAIN 1.112983215e+06
static float xv[NZEROS+1], yv[NPOLES+1];
static void filterloop()
{ for (;;)
    { xv[0] = xv[1]; xv[1] = xv[2]; xv[2] = xv[3]; xv[3] = xv[4];
      xv[4] = next input value / GAIN;
    }
}

```

```

yv[0] = yv[1]; yv[1] = yv[2]; yv[2] = yv[3]; yv[3] = yv[4];

yv[4] = (xv[0] + xv[4]) + 4 * (xv[1] + xv[3]) + 6 * xv[2]
        + (-0.8485559993 * yv[0]) + ( 3.5335352195 * yv[1])
        + (-5.5208191366 * yv[2]) + ( 3.8358255406 * yv[3]);

next output value = yv[4];
}
}
*/

void __fastcall TForm_principal::Configurao1Click(TObject *Sender)
{
    Form_config_porta->ShowModal();
}

//-----

void TForm_principal::ReturnMethod(AnsiString NewData)
{
    dados_sinal = dados_sinal + NewData;

    /*    if(tamanho_vetor < 1000)
        {
            try
                {*/

    /*        if(dados_sinal[tamanho_vetor] < 0)
            {
                dados_sinal[tamanho_vetor] = dados_sinal[tamanho_vetor] * -1;
            }

            uc_auxiliar = dados_sinal[tamanho_vetor];
            Series1->AddXY(tamanho_vetor, uc_auxiliar,
                "", clGreen);
        }
        catch(...)
        {

```

```

        ;
    }
}
else
{
    Serial->Disconnect();//Desconecta para
        //poder efetuar reinicializacoes
    dados_sinal = "";//Limpa string
    Series1->Clear();
    Serial->Connect();
}*/
}
void __fastcall TForm_principal::FormDestroy(TObject *Sender)
{
    try
    {
        Serial->Disconnect();
    }
    catch(...)
    {
        ;
    }
}
void __fastcall TForm_principal::Button_iniciaClick(TObject *Sender)
{
    try
    {
        Serial->SetDeviceName(porta);//Define a porta a ser usada
        Serial->Connect();//Efetua a conexao
        if(Serial->Connect())//Verifica se a porta foi conectada

```

```

    {
        Shape1->Brush->Color = clRed;

        Serial->SetBaudRate(velocidade);

        Serial->SetByteSize(8);

        Serial->SetParity(NOPARITY);

        Serial->SetStopBits(ONESTOPBIT);

        Panel3->Caption = "";

        Panel4->Caption = "";

        Memo1->Clear();

        Button_inicia->Enabled = False;

        Panel3->Caption = TimeToStr(Time());

        Button_finaliza->Enabled = True;

    }

    else

    {

        Shape1->Brush->Color = clWhite;

    }

    Series1->Clear();

}

catch(...)

{

    ;

}

//    Timer2->Enabled = True;

}

void __fastcall TForm_principal::Button_finalizaClick(TObject *Sender)

{

    int i_m;

    int i_largura_fft;

    i_m = 10;

```

```

i_largura_fft = 1 << i_m;

NUMERO_COMPLEXO *amostra;//Cria estrutura com

    //o nome de amostra

Timer2->Enabled = False;

try

{

    Serial->Disconnect();

    Shape1->Brush->Color = clWhite;

    Button_finaliza->Enabled = False;

    Button_inicia->Enabled = True;

    Panel4->Caption = TimeToStr(Time());

    tamanho_vetor = dados_sinal.Length();

    Memo1->Text = IntToStr(tamanho_vetor);

    amostra = (NUMERO_COMPLEXO *) calloc(i_largura_fft, sizeof(NUMERO_COMPLEXO));

    for(int contador = 1; contador <= tamanho_vetor; contador++)

    {

        i_auxiliar = dados_sinal[contador];

        if(i_auxiliar <= 0)

        {

            i_auxiliar = (i_auxiliar * -5);

        }

        else

        {

            i_auxiliar = i_auxiliar * 5;

        }

        if(contador <= 499)

        {

            amostra[contador].real = i_auxiliar;

        }

        Series1->AddXY(contador,

```

```

        i_auxiliar,
        "", clBlue);
    }
    funcao_fft(amostra, i_m);
    for(int contador = 0; contador <= 499; contador++)
    {
        Series2->AddXY(contador,
            (pow(pow(amostra[contador].real, 2) +
                pow(amostra[contador].imaginario, 2), 0.5) / 1000),
            "", clRed);
    }
    dados_sinal = "";
}
catch(...)
{
    ;
}
}
void __fastcall TForm_principal::RadioButon3Click(TObject *Sender)
{
    Serial->Disconnect();
    Timer2->Enabled = False;
}
void __fastcall TForm_principal::Timer2Timer(TObject *Sender)
{
    int i_m;
    int i_largura_fft;
    i_m = 10;
    i_largura_fft = 1 << i_m;
    NUMERO_COMPLEXO *amostra;//Cria estrutura com o nome de amostra

```

```

amostra = (NUMERO_COMPLEXO *) calloc(i_largura_fft, sizeof(NUMERO_COMPLEXO));

Series1->Clear();

Series2->Clear();

tamanho_vetor = dados_sinal.Length();

for(int contador = 1; contador <= tamanho_vetor; contador++)
{
    try
    {
        uc_auxiliar = i_ganho_sinal * dados_sinal[contador];

        Series1->AddXY(contador, uc_auxiliar,
            "", clGreen);

        if(contador <= 499)
        {
            switch(i_tipo_janelamento)
            {
                case 0://Nenhuma ou Retangular
                    amostra[contador].real = uc_auxiliar;

                    break;

                case 1://Hanning
                    amostra[contador].real = uc_auxiliar *
                        (0.5 - 0.5 * (cos((2*M_PI*contador)/tamanho_vetor)));

                    break;

                case 2://Hamming
                    amostra[contador].real = uc_auxiliar *
                        (0.54 - 0.46 * (cos((2*M_PI*contador)/tamanho_vetor)));

                    break;

                case 3://Triangular
                    amostra[contador].real = uc_auxiliar *
                        (1 - ((2*M_PI*contador)/(tamanho_vetor)));

                    break;
            }
        }
    }
}

```

```

        default :
            break;
        }
    }
}
catch(...)
{
    ;
}
}
funcao_fft(amostra, i_m);
for(int contador = 0; contador <= 499; contador++)
{
    f_magnitude_fft = (pow(pow(amostra[contador].real, 2) +
        pow(amostra[contador].imaginario, 2), 0.5) / 1000) *
        i_ganho_fft;

    if(((contador*1.70) >= 39.1) && ((contador*1.70) <= 40.9))
    {
        Memo1->Text = FloatToStr(f_magnitude_fft);
        if(f_magnitude_fft > f_valor_inferior)
        {
            try
            {
                Serial->TransmitData("a");
            }
            catch(...)
            {
                ;
            }
        }
    }
}
}

```

```

        }
    }
    else
    {
        try
        {
            Serial->TransmitData("d");//Ativa motor de fechamento do braço
        }
        catch(...)
        {
            ;
        }
    }
}

Series2->AddXY(contador * 1.70, f_magnitude_fft, "", clRed);
}

dados_sinal = "";

}

void __fastcall TForm_principal::MaskEdit_ganhoKeyPress(TObject *Sender,
char &Key)
{
    if(Key == 13)
    {
        try
        {
            i_ganho_sinal = StrToInt(MaskEdit_ganho->Text);
            RadioButton1->Checked = False;
            RadioButton2->Checked = False;

```

```

    }
    catch(...)
    {
        i_ganho_sinal = 1;
    }
}

void __fastcall TForm_principal::MaskEdit_ganho_fftKeyPress(
    TObject *Sender, char &Key)
{
    if(Key == 13)
    {
        try
        {
            i_ganho_fft = StrToInt(MaskEdit_ganho_fft->Text);
            RadioButton1->Checked = False;
            RadioButton2->Checked = False;
        }
        catch(...)
        {
            i_ganho_fft = 1;
        }
    }
}

void __fastcall TForm_principal::Timer1Timer(TObject *Sender)
{
    /*    if(i_status_mao == 1)
        {

```

```

        i_status_mao = 0;

        try
        {
//            Serial->TransmitData("a");
        }

        catch(...)
        {
            ;
        }
    }
else
    {
        i_status_mao = 1;

        try
        {
//            Serial->TransmitData("d");
        }

        catch(...)
        {
            ;
        }
    }*/

    Label_estado_serial_1->Caption = s_estado_serial_1;
}

void __fastcall TForm_principal::RadioButton_janelamento_nenhumClick(
    TObject *Sender)
{
    i_tipo_janelamento = 0;//Determina o que não houvera
        //janelamento

```

```

}

void __fastcall TForm_principal::RadioButton5Click(TObject *Sender)
{
    i_tipo_janelamento = 1;//Determina que o janelamento sera
        //no formato Hanning
        //  $w[n] = 0.5 - 0.5 * \cos(2\pi n / N)$ 
}

void __fastcall TForm_principal::RadioButton6Click(TObject *Sender)
{
    i_tipo_janelamento = 2;//Determina que o janelamento sera
        //no formato Hamming
        //  $w[n] = 0.54 - 0.46 * \cos(2\pi n / N)$ 
}

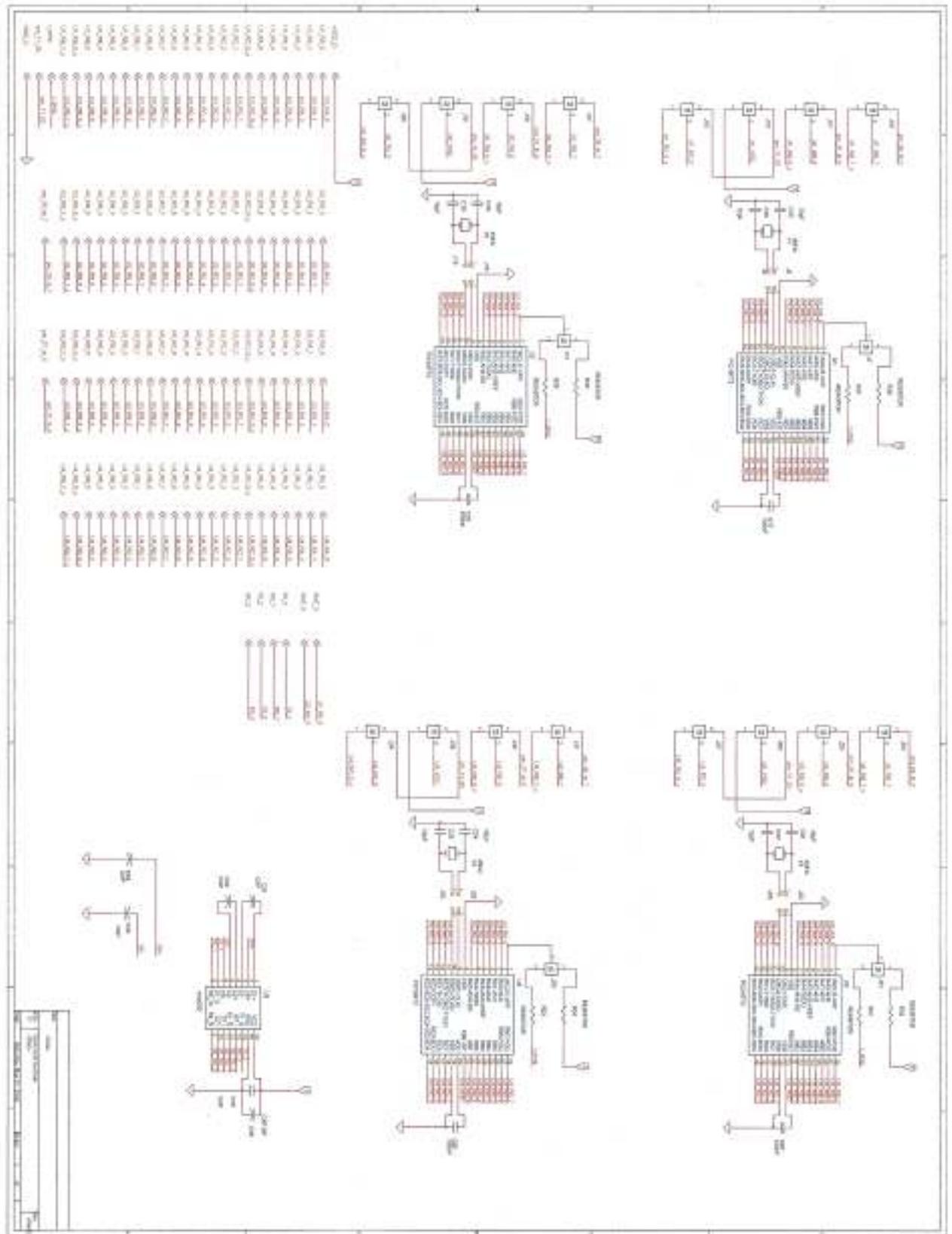
void __fastcall TForm_principal::RadioButton4Click(TObject *Sender)
{
    i_tipo_janelamento = 3;//Determina que o janelamento sera
        //no formato Triangular
}

void __fastcall TForm_principal::Edit_valor_inferiorKeyPress(
    TObject *Sender, char &Key)
{
    if(Key == 13)
    {
        try
        {
            f_valor_inferior = StrToFloat(Edit_valor_inferior->Text);
            Label_valor_inferior->Caption = FormatFloat("0.00", f_valor_inferior);
        }
        catch (...)
        {
            // Erro de conversão de string para float
        }
    }
}

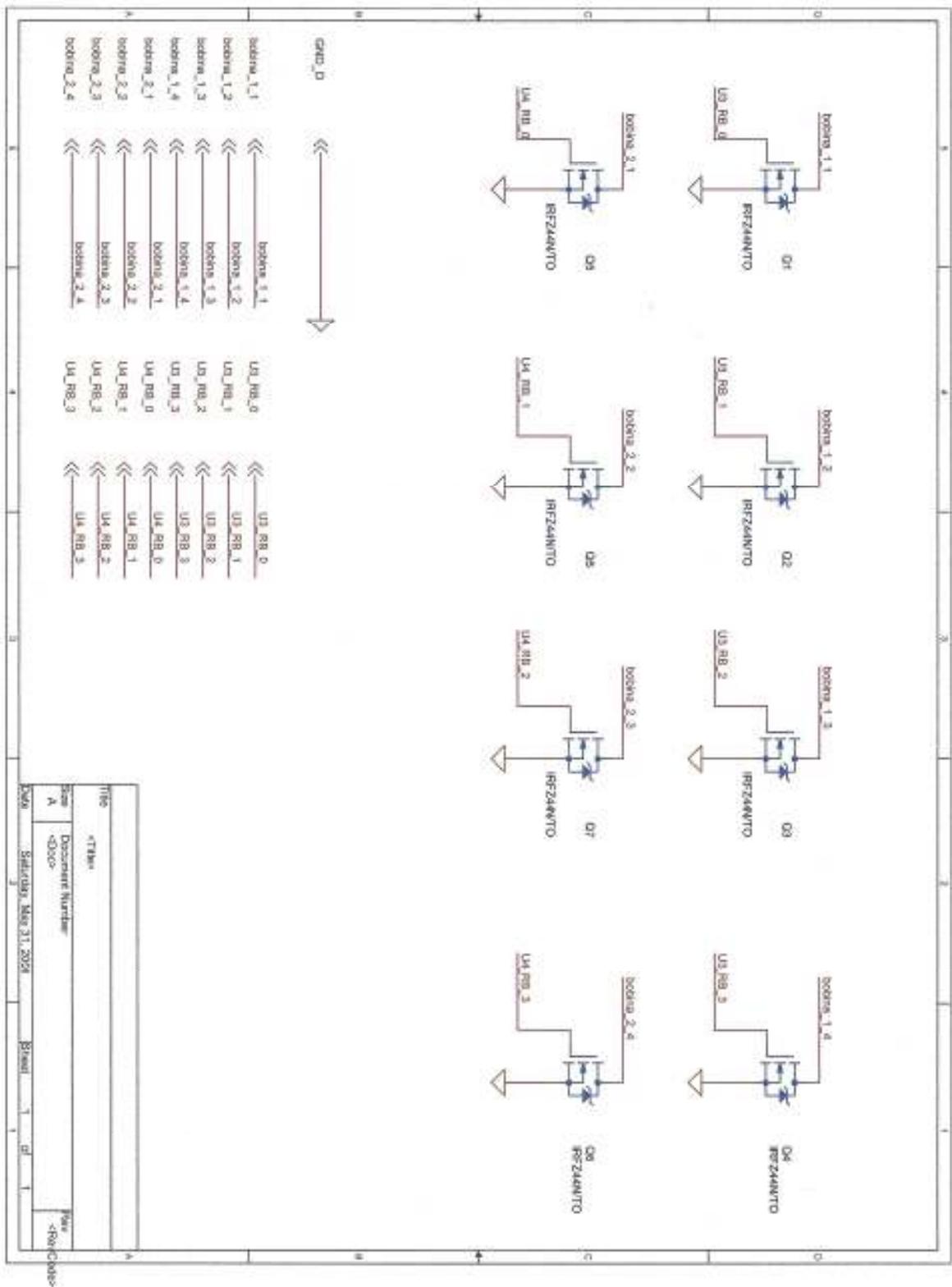
```

```
    }  
    catch(...)  
    {  
        f_valor_inferior = 0.2;  
        Label_valor_inferior->Caption = FormatFloat("0.00", f_valor_inferior);  
    }  
}  
}
```

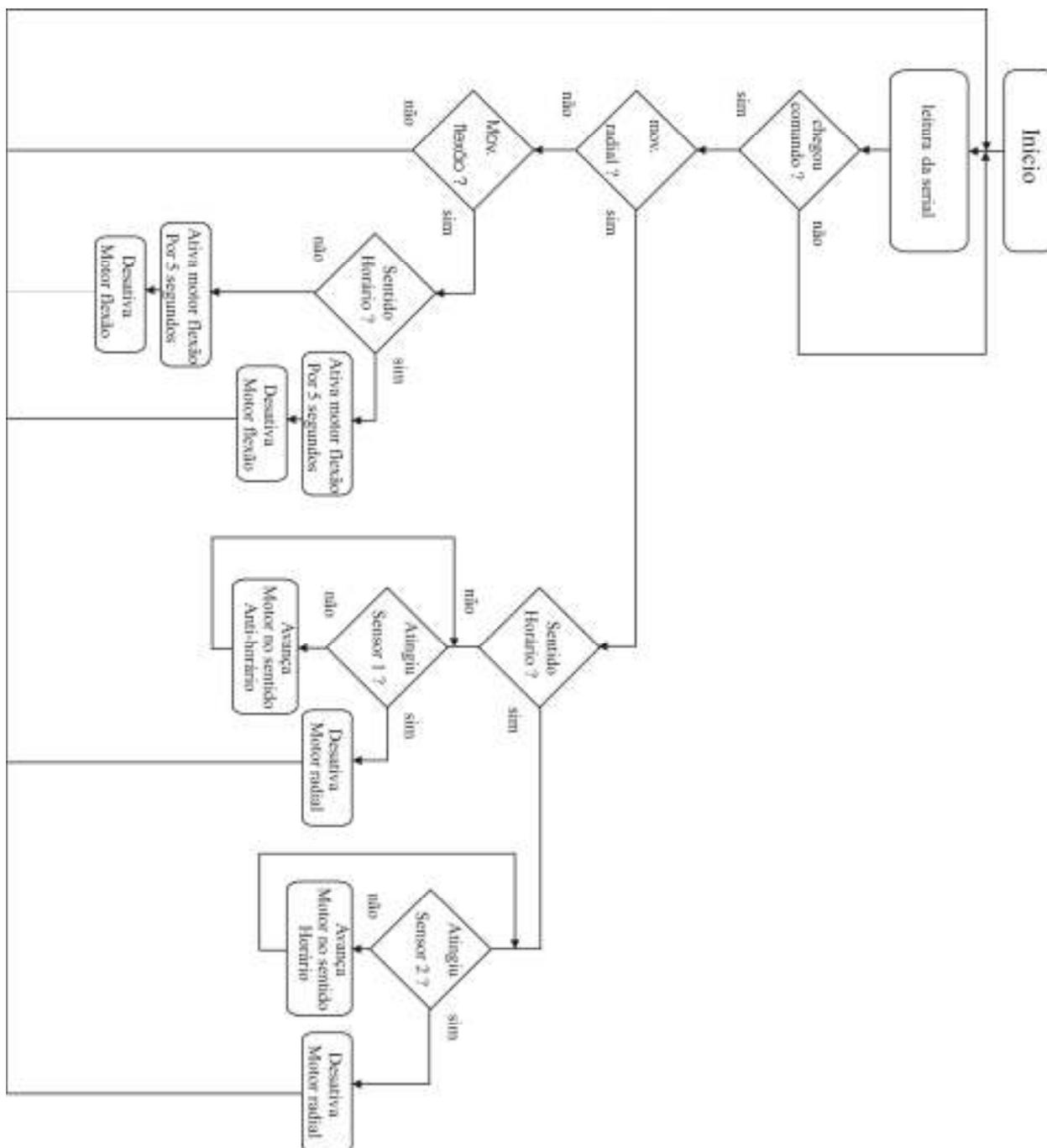
APÊNDICE D – ESQUEMA ELÉTRICO – INTERFACE DE AQUISIÇÃO – DIGITAL



APÊNDICE F – ESQUEMA ELÉTRICO – INTERFACE DE POTÊNCIA – DRIVERS



**APÊNDICE G – FLUXOGRAMA DE TESTES NA INTERFACE
DE POTÊNCIA E MÃO MECÂNICA**



**APÊNDICE H – FONTE DO PROGRAMA DE TESTES
DA INTERFACE DE POTÊNCIA**

```
/*
```

```
TRABALHO DE CONCLUSAO DE CURSO  
ENGENHARIA ELETRICA
```

```
POR : Fabio de Araujo Bairros  
NOME : TESTE DA INTERFACE DE POTENCIA E MOTORES  
DISCIPLINA : TCC I E TCC II  
CLIENTE : ULBRA  
DATA : 04/05/2008
```

```
OBS: Os sensores da mao ainda nao estao implementados, apenas os sensores do movimento radial
```

```
*/
```

```
#include <16f72.h>
```

```
//Inclui cabeçalho do
```

```
componente
```

```
#device adc=8;//DEFINE COMO 8 BITS PARA O AD
```

```
#fuses HS, NOWDT, PUT, NOBROWNOUT, NOPROTECT, XT  
//Define estado dos fusíveis
```

```
//NOLVP
```

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

```
//Define o valor do clock
```

```
#include <rs232.c>
```

```
//rotinas de que usaram o canal serial
```

```
//RS-232
```

```
//#use fixed_io (a_outputs = pin_a0, pin_a1, pin_a2, pin_a3, pin_a4, pin_a5)  
//Define porto A como saída
```

```
#use fixed_io (b_outputs = pin_b0, pin_b1, pin_b2, pin_b3, pin_b4, pin_b5, pin_b6, pin_b7)  
//Define o Porto B como saída
```

```
//#use fixed_io (c_outputs = pin_c0, pin_c1, pin_c3, pin_c5)  
//RC3 = SCLK
```

```
//RC5 =
```

```
SDO
```

```
int contador;
```

```

int          valor_adc;          //RECEBE O VALOR DA CONVERÇÃO
DO AD
int          i_dado_recebido;    //DADOS RECEBIDO PELA SERIAL

int          tempo_motor;       //TEMPO DE DELAY PARA O PASSO DO
MOTOR
int          valor_teclado;     //RECEBE O VALOR DO TECLADO
                                //0 -> nenhuma tecla
                                //1 -> tecla direita
                                //2 -> tecla esquerda
int          valor_motor;       //GUARDA O VALOR ATUAL O MOTOR

boolean      status_sentido;     //INDICA EM QUE SENTIDO O
MOTOR ESTA GIRANTO NESTE MOMENTO
                                //0 -> HORARIO
                                //1 -> ANTI-HORARIO

boolean      leitura_sensor;    //RECEBE O VALOR DE UM DETERMINADO
PINO
boolean      sensor_direita;    //GUARDA ESTADO DO SENSOR DA
DIREITA - ATIVADO PELO MOVIMENTO NO SENTIDO HORARIO
boolean      sensor_esquerda;  //GUARDA ESTADO DO SENSOR DA
ESQUERDA - ATIVADO PELO MOVIMENTO NO SENTIDO ANTI-HORARIO
boolean      tecla_direita;     //GUARDA O VALOR DA TECLA LIGADA AO
PINO C2 -> TECLA DIREITA
boolean      tecla_esquerda;   //GUARDA O VALOR DA TECLA
LIGADA AO PINO C3 -> TECLA ESQUERDA
boolean      tecla_1;          //TECLAD DA DIREITA
boolean      tecla_2;          //TECLA DA ESQUERDA

#####
#####
#####
#####
//DESLIGAR MOTOR PUNHO -> COLOCA TODOS OS PINO EM ZERO
void para_motor_punho(void)
{
    output_low(pin_b0);
    output_low(pin_b1);
    output_low(pin_b2);
    output_low(pin_b3);
    delay_ms(10);
}
#####
#####
#####
#####
//DESLIGAR MOTOR MAO -> COLOCA TODOS OS PINO EM ZERO

```

```

void para_motor_mao(void)
{
    output_low(pin_a0);
    output_low(pin_a1);
    output_low(pin_a2);
    output_low(pin_a3);
    delay_ms(10);
}

#####
#####
#####
#####
//LEITURA DOS SENSORES
void le_sensores_punho(void)
{
    sensor_direita = input(pin_c1);
    sensor_esquerda = input(pin_c0);
}

#####
#####
#####
#####
//LEITURA DOS SENSORES
void le_sensores_mao(void)
{
    sensor_direita = input(pin_c1);
    sensor_esquerda = input(pin_c0);
}

#####
#####
#####
#####
//ACIONAMENTO DO MOTOR QUE EFETUA O GIRO DO PUNHO -> SENTIDO
HORARIO
void motor_punho_horario(void)
{
    valor_motor = 8; //Inicia com valor 8
    para_motor_punho(); //Zera as
saida das bobinas

    le_sensores_punho(); //Le os
sensores antes de entrar no laço

    while(!sensor_esquerda)

```

```

//          while(1)
//          {
//              output_b(valor_motor);
//              delay_ms(tempo_motor);
//              para_motor_punho();
//              valor_motor = valor_motor >> 1;           //Desloca o valor para
direita uma vez
//              if(valor_motor <= 0)
//              {
//                  valor_motor = 8;
//              }
//              le_sensores_punho();
//          }
//
//          para_motor_punho();
//
//
//
//#####
//#####
//#####
//#####
//ACIONAMENTO DO MOTOR QUE EFETUA O GIRO DO PUNHO -> SENTIDO ANTI-
HORARIO
void motor_punho_anti_horario(void)
{
//          valor_motor = 1;           //Inicia com valor 8
//          para_motor_punho();       //Zera as
saida das bobinas
//          le_sensores_punho();     //Le os
sensores antes de entrar no laco
//
//          while(!sensor_direita)
//          while(1)
//          {
//              output_b(valor_motor);
//              delay_ms(tempo_motor);
//              para_motor_punho();
//              valor_motor = valor_motor << 1;       //Desloca o valor para
esquerda uma vez
//              if(valor_motor >= 16)
//              {
//                  valor_motor = 1;
//              }
//              le_sensores_punho();
//          }

```

```

    para_motor_punho());

}
#####
#####
#####
#####
//ACIONAMENTO DO MOTOR QUE EFETUA O FECHAMENTO DA MAO -> SENTIDO
HORARIO
void motor_mao_horario(void)
{

    valor_motor = 8; //Inicia com valor 8
    para_motor_mao(); //Zera as
saida das bobinas

    //le_sensores_mao(); //Le os
sensores antes de entrar no laco

    while(!sensor_esquerda)
    {
        output_a(valor_motor); //No porto A este
localizado os sinais do motor
        delay_ms(tempo_motor);
        para_motor_mao();
        valor_motor = valor_motor >> 1; //Desloca o valor para
direita uma vez
        if(valor_motor <= 0)
        {
            valor_motor = 8;
        }
        le_sensores_mao();
    }

    para_motor_mao();

}
#####
#####
#####
#####
//ACIONAMENTO DO MOTOR QUE EFETUA O FECHAMENTO DA MAO -> SENTIDO
ANTI-HORARIO
void motor_mao_anti_horario(void)
{

    valor_motor = 1; //Inicia com valor 8
    para_motor_mao(); //Zera as
saida das bobinas

```

```

        //le_sensores_mao(); //Le os
sensores antes de entrar no laço

        while(!sensor_direita)
        {
            output_a(valor_motor); //No porto A esta
localizado os sinais do motor
            delay_ms(tempo_motor);
            para_motor_mao();
            valor_motor = valor_motor << 1; //Desloca o valor para
esquerda uma vez
            if(valor_motor >= 16)
            {
                valor_motor = 1;
            }
            le_sensores_mao();
        }

        para_motor_mao();

    }

#####
#####
#####
#####
/*
FUNCAO PRINCIPAL - CORPO DO PROGRAMA
*/

void main(void)
{

    set_tris_c(0xff);
//Todos os pinos do porto C como entrada

    rs232_inicializa();
//Inicializa canal serial com parametros de velocidade, controle

    para_motor_mao();
//Chama funcao para desligar motor
    para_motor_punho();
//Chama funcao para desligar motor

    tempo_motor = 100;

    while(1)
    {

```

```

i_dado_recebido = rs232_recebe(); //Aguarda o dado
pela serial

switch(i_dado_recebido)//testa os comandos recebidos da serial
{
    case 'a':
        output_high(pin_b7);
        motor_punho_anti_horario();// Ativa motor radial
        break;
    case 'd':
        output_low(pin_b7);
        motor_punho_horario();//Ativa motor radial ate
        break;
    case '3':
        output_high(pin_b7);
        motor_mao_horario();//Ativa motor flexao por 5
        delay(5000);
        break;
    case '4':
        output_low(pin_b7);
        motor_mao_anti_horario();//Ativa motor flexao
        delay(5000);
        break;
    default :
        break;
}
}
}

```

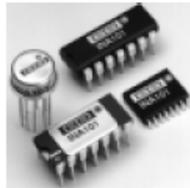
APÊNDICE I – CRONOGRAMA DE ATIVIDADES

Atividades / Período		Jan/2008	Fev/2008	Mar/2008	Abr/2008	Mai/2008	Jun/2008
Pesquisa	Materiais	x					
	Componentes		x		x		x
Projeto	Hardware		x		x		
	Software		x		x		
Execução	Hardware			x			x
	Software			x		x	
Documentação	Monografia					x	x
	Relatório					x	x
Média horas/dia							
4 horas por dia							

ANEXOS

ANEXO A

DATA-SHEET – AMPLIFICADOR DE INSTRUMENTAÇÃO - INA 101
 – TEXAS INSTRUMENTS – BURR-BROWN



INA101

High Accuracy
 INSTRUMENTATION AMPLIFIER

FEATURES

- LOW DRIFT: 0.25 μ V/ $^{\circ}$ C max
- LOW OFFSET VOLTAGE: 25 μ V max
- LOW NONLINEARITY: 0.002%
- LOW NOISE: 13nV/ $\sqrt{\text{Hz}}$
- HIGH CMR: 106dB AT 60Hz
- HIGH INPUT IMPEDANCE: 10¹⁰ Ω
- 14-PIN PLASTIC, CERAMIC DIP, SOL-16, AND TO-100 PACKAGES

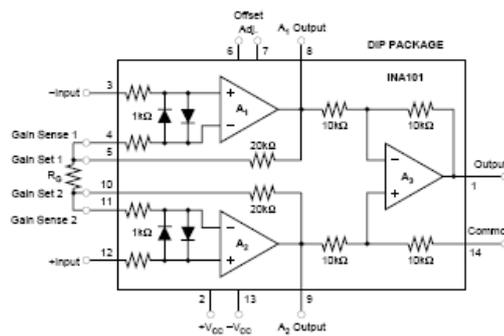
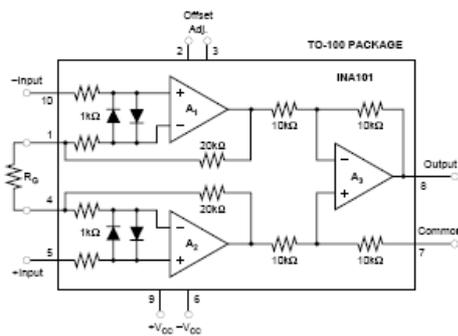
APPLICATIONS

- STRAIN GAGES
- THERMOCOUPLES
- RTDs
- REMOTE TRANSDUCERS
- LOW-LEVEL SIGNALS
- MEDICAL INSTRUMENTATION

DESCRIPTION

The INA101 is a high accuracy instrumentation amplifier designed for low-level signal amplification and general purpose data acquisition. Three precision op amps and laser-trimmed metal film resistors are integrated on a single monolithic integrated circuit.

The INA101 is packaged in TO-100 metal, 14-pin plastic and ceramic DIP, and SOL-16 surface-mount packages. Commercial, industrial and military temperature range models are available.



ANEXO B

DATA-SHEET – MICROCONTROLADOR – PIC 16F72 – MICROCHIP



PIC16F72

28-Pin, 8-Bit CMOS FLASH MCU with A/D Converter

Device Included:

- PIC16F72

High Performance RISC CPU:

- Only 35 single word instructions to learn
- All single cycle instructions except for program branches, which are two-cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- 2K x 14 words of Program Memory,
128 x 8 bytes of Data Memory (RAM)
- Pinout compatible to PIC16C72/72A and
PIC16F872
- Interrupt capability
- Eight-level deep hardware stack
- Direct, Indirect and Relative Addressing modes

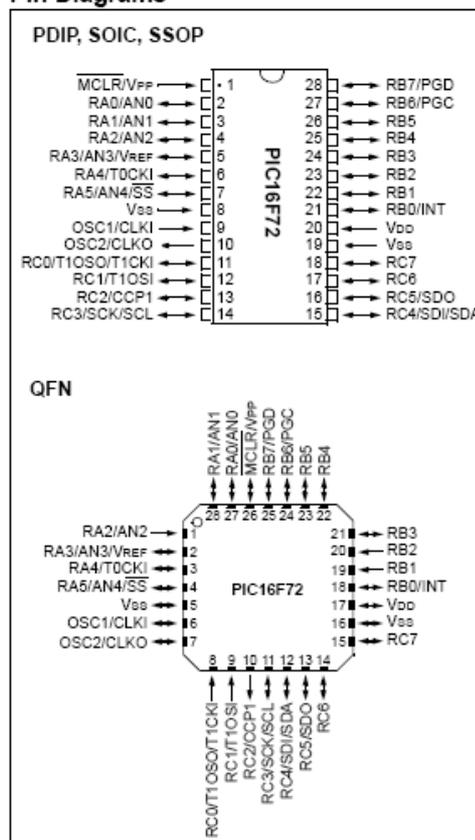
Peripheral Features:

- High Sink/Source Current: 25 mA
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during SLEEP via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Capture, Compare, PWM (CCP) module
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- 8-bit, 5-channel analog-to-digital converter
- Synchronous Serial Port (SSP) with
SPI™ (Master/Slave) and I²C™ (Slave)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

CMOS Technology:

- Low power, high speed CMOS FLASH technology
- Fully static design
- Wide operating voltage range: 2.0V to 5.5V
- Industrial temperature range
- Low power consumption:
 - < 0.6 mA typical @ 3V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagrams



Special Microcontroller Features:

- 1,000 erase/write cycle FLASH program memory typical
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- In-Circuit Serial Programming™ (ICSP™) via 2 pins
- Processor read access to program memory

ANEXO C

DATA-SHEET – INTERFACE RS-232 – HIN 232 – INTERSIL


**HIN202E, HIN206E, HIN207E, HIN208E,
HIN211E, HIN213E, HIN232E**

Data Sheet

November 4, 2005

FN4315.16

**±15kV, ESD-Protected, +5V Powered,
RS-232 Transmitters/Receivers**

The HIN202E, HIN206E, HIN207E, HIN208E, HIN211E, HIN213E, HIN232E family of RS-232 transmitters/receivers interface circuits meet all EIA high-speed RS-232E and V.28 specifications, and are particularly suited for those applications where ±12V is not available. A redesigned transmitter circuit improves data rate and slew rate, which makes this suitable for ISDN and high speed modems. The transmitter outputs and receiver inputs are protected to ±15kV ESD (Electrostatic Discharge). They require a single +5V power supply and feature onboard charge pump voltage converters which generate +10V and -10V supplies from the 5V supply. The family of devices offers a wide variety of high-speed RS-232 transmitter/receiver combinations to accommodate various applications (see Selection Table).

The HIN206E, HIN211E and HIN213E feature a low power shutdown mode to conserve energy in battery powered applications. In addition, the HIN213E provides two active receivers in shutdown mode allowing for easy "wake-up" capability.

The drivers feature true TTL/CMOS input compatibility, slew rate-limited output, and 300Ω power-off source impedance. The receivers can handle up to ±30V input, and have a 3kΩ to 7kΩ input impedance. The receivers also feature hysteresis to greatly improve noise rejection.

Features

- Pb-Free Plus Anneal Available (RoHS Compliant)
- High Speed ISDN Compatible 230kbits/s
- ESD Protection for RS-232 I/O Pins to ±15kV (IEC61000)
- Meets All RS-232E and V.28 Specifications
- Requires Only 0.1μF or Greater External Capacitors
- Two Receivers Active in Shutdown Mode (HIN213E)
- Requires Only Single +5V Power Supply
- Onboard Voltage Doubler/Inverter
- Low Power Consumption (Typ) 5mA
- Low Power Shutdown Function (Typ) 1μA
- Three-State TTL/CMOS Receiver Outputs
- Multiple Drivers
 - ±10V Output Swing for +5V Input
 - 300Ω Power-Off Source Impedance
 - Output Current Limiting
 - TTL/CMOS Compatible
- Multiple Receivers
 - ±30V Input Voltage Range
 - 3kΩ to 7kΩ Input Impedance
 - 0.5V Hysteresis to Improve Noise Rejection

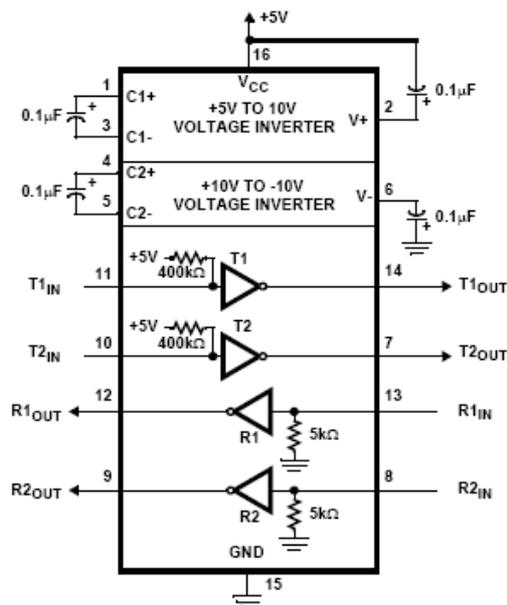
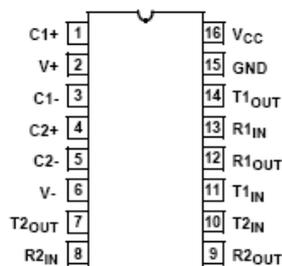
Applications

- Any System Requiring High-Speed RS-232 Communications Port
 - Computer - Portable, Mainframe, Laptop
 - Peripheral - Printers and Terminals
 - Instrumentation, UPS
 - Modems, ISDN Terminal Adaptors

Selection Table

PART NUMBER	POWER SUPPLY VOLTAGE	NUMBER OF RS-232 DRIVERS	NUMBER OF RS-232 RECEIVERS	NUMBER OF 0.1μF EXTERNAL CAPACITORS	LOW POWER SHUTDOWN/TTL THREE-STATE	NUMBER OF RECEIVERS ACTIVE IN SHUTDOWN
HIN202E	+5V	2	2	4 Capacitors	No/No	0
HIN206E	+5V	4	3	4 Capacitors	Yes/Yes	0
HIN207E	+5V	5	3	4 Capacitors	No/No	0
HIN208E	+5V	4	4	4 Capacitors	No/No	0
HIN211E	+5V	4	5	4 Capacitors	Yes/Yes	0
HIN213E	+5V	4	5	4 Capacitors	Yes/Yes	2
HIN232E	+5V	2	2	4 Capacitors	No/No	0

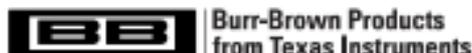
HIN202E, HIN206E, HIN207E, HIN208E, HIN211E, HIN213E, HIN232E
Pinouts (Continued)

 HIN232E (PDIP, SOIC, SSOP, TSSOP)
 TOP VIEW

Pin Descriptions

PIN	FUNCTION
V _{CC}	Power Supply Input 5V ±10%, (5V ±5% HIN207E).
V+	Internally generated positive supply (+10V nominal).
V-	Internally generated negative supply (-10V nominal).
GND	Ground Lead. Connect to 0V.
C1+	External capacitor (+ terminal) is connected to this lead.
C1-	External capacitor (- terminal) is connected to this lead.
C2+	External capacitor (+ terminal) is connected to this lead.
C2-	External capacitor (- terminal) is connected to this lead.
T _{IN}	Transmitter Inputs. These leads accept TTL/CMOS levels. An internal 400kΩ pull-up resistor to V _{CC} is connected to each lead.
T _{OUT}	Transmitter Outputs. These are RS-232 levels (nominally ±10V).
R _{IN}	Receiver Inputs. These inputs accept RS-232 input levels. An internal 5kΩ pull-down resistor to GND is connected to each input.
R _{OUT}	Receiver Outputs. These are TTL/CMOS levels.
$\overline{\text{EN}}$, EN	Receiver Enable Input. With $\overline{\text{EN}} = 5V$ (HIN213E EN=0V), the receiver outputs are placed in a high impedance state.
SD, $\overline{\text{SD}}$	Shutdown Input. With SD = 5V (HIN213E $\overline{\text{SD}} = 0V$), the charge pump is disabled, the receiver outputs are in a high impedance state (except R4 and R5 of HIN213E) and the transmitters are shut off.
NC	No Connect. No connections are made to these leads.

ANEXO D

DATA-SHEET – AMPLIFICADOR OPERACIONAL – OPA 4227
– TEXAS INSTRUMENTS – BURR BROWN



OPA227
OPA2227
OPA4227
OPA228
OPA2228
OPA4228

SBO511DA – MAY 1995 – REVISED JANUARY 2005

High Precision, Low Noise OPERATIONAL AMPLIFIERS

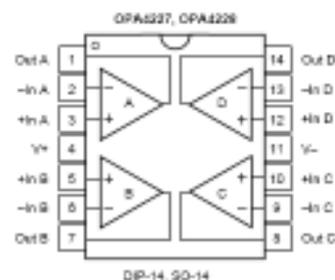
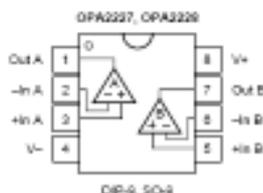
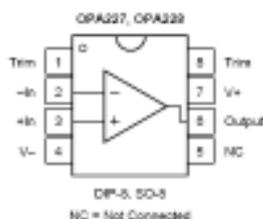
FEATURES

- **LOW NOISE:** $3\text{nV}/\sqrt{\text{Hz}}$
- **WIDE BANDWIDTH:**
OPA227: 8MHz, 2.3V/ μs
OPA228: 33MHz, 10V/ μs
- **SETTLING TIME:** 5 μs
(significant improvement over OP-27)
- **HIGH CMRR:** 138dB
- **HIGH OPEN-LOOP GAIN:** 160dB
- **LOW INPUT BIAS CURRENT:** 10nA max
- **LOW OFFSET VOLTAGE:** 75 μV max
- **WIDE SUPPLY RANGE:** $\pm 2.5\text{V}$ to $\pm 18\text{V}$
- **OPA227 REPLACES** OP-27, LT1007, MAX427
- **OPA228 REPLACES** OP-37, LT1037, MAX437
- **SINGLE, DUAL, AND QUAD VERSIONS**

APPLICATIONS

- DATA ACQUISITION
- TELECOM EQUIPMENT
- GEOPHYSICAL ANALYSIS
- VIBRATION ANALYSIS
- SPECTRAL ANALYSIS
- PROFESSIONAL AUDIO EQUIPMENT
- ACTIVE FILTERS
- POWER SUPPLY CONTROL

SPICE model available for OPA227 at www.ti.com



DESCRIPTION

The OPA227 and OPA228 series op amps combine low noise and wide bandwidth with high precision to make them the ideal choice for applications requiring both ac and precision dc performance.

The OPA227 is unity-gain stable and features high slew rate (2.3V/ μs) and wide bandwidth (8MHz). The OPA228 is optimized for closed-loop gains of 5 or greater, and offers higher speed with a slew rate of 10V/ μs and a bandwidth of 33MHz.

The OPA227 and OPA228 series op amps are ideal for professional audio equipment. In addition, low quiescent current and low cost make them ideal for portable applications requiring high precision.

The OPA227 and OPA228 series op amps are pin-for-pin replacements for the industry standard OP-27 and OP-37 with substantial improvements across the board. The dual and quad versions are available for space savings and per-channel cost reduction.

The OPA227, OPA228, OPA2227, and OPA2228 are available in DIP-8 and SO-8 packages. The OPA4227 and OPA4228 are available in DIP-14 and SO-14 packages with standard pin configurations. Operation is specified from -40°C to $+85^{\circ}\text{C}$.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

**TEXAS
INSTRUMENTS**
www.ti.com

Copyright © 1995-2005, Texas Instruments Incorporated

