



UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA

GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

**DESENVOLVIMENTO DE UMA PLATAFORMA  
EMBARCADA PARA POSICIONAMENTO DE MOTORES  
COM VALIDAÇÃO EM UM SISTEMA FOTOVOLTAICO**

**ERNANDO PASSOS BATISTA JUNIOR**

Cruz Das Almas - 2015

UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA

GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

**DESENVOLVIMENTO DE UMA PLATAFORMA  
EMBARCADA PARA POSICIONAMENTO DE  
MOTORES COM VALIDAÇÃO EM UM SISTEMA  
FOTOVOLTAICO**

Trabalho de Conclusão de Curso apresentado à  
Universidade Federal do Recôncavo como parte dos  
requisitos para obtenção do título de Engenheiro de  
Computação.

Orientador (a): Igor Dantas dos Santos Miranda

**ERNANDO PASSOS BATISTA JUNIOR**

Cruz Das Almas - 2015

UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA

GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

**DESENVOLVIMENTO DE UMA PLATAFORMA  
EMBARCADA PARA POSICIONAMENTO DE  
MOTORES COM VALIDAÇÃO EM UM SISTEMA  
FOTOVOLTAICO**

Aprovada em: 20 de maio de 2015

EXAMINADORES:

Prof. Igor Dantas dos Santos Miranda

Prof. Dr. José Valentim dos Santos Filho

Prof. Msc. Acbal Rucas Andrade Achy

**ERNANDO PASSOS BATISTA JUNIOR**

Cruz Das Almas - 2015

## **Agradecimentos**

*Agradeço primeiramente a Deus que está sempre em minha mente, com certeza sem ele eu não estaria nesta importante etapa da vida.*

*A Dona Dulcimar, mãe querida, que tanto torceu por mim e soube me apoiar nos momentos de dificuldade.*

*Ao meu grande pai, em memória, pois me ensinou muito para me tornar um homem de caráter.*

*A toda minha família, sobretudo, ao meu tio Reinaldo Batista por estar sempre me ajudando durante minha formação.*

*A Alana, pela compreensão e companhia.*

*Agradeço ao Professor Igor pela orientação atuante, e por me mostrar o mundo dos sistemas embarcados.*

*Agradeço aos meus amigos de graduação, Alex, Eliabe, Joel e Marcos Bião e ao corpo docente do curso.*

*Sou muito grato por todos que colaboraram para a realização deste sonho.*



UNIVERSIDADE FEDERAL DO RECONCAVO DA BAHIA

**DESENVOLVIMENTO DE UMA PLATAFORMA EMBARCADA PARA  
POSICIONAMENTO DE MOTORES COM VALIDAÇÃO EM UM SISTEMA  
FOTOVOLTAICO**

**RESUMO**

Os motores elétricos desempenham um papel fundamental na sociedade moderna, estando presente nos mais variados setores, tais como: a indústria, a agricultura, o ambiente acadêmico, o comércio, geração de energia, entre outros. Um fator limitante na utilização de motores em projetos de Pesquisas e Desenvolvimento (P&D), ocorre devido a dificuldades encontradas na sua configuração e controle. Existe uma carência em soluções tecnológicas que proporcionem ao usuário uma maior experiência na utilização de motores elétricos nas mais diversificadas esferas. Dentre as inúmeras áreas de conhecimento que poderiam fazer uso de um sistema intuitivo para posicionamento de motores, destaca-se o setor de energia solar devido a sua disponibilidade e abundância. Sistemas fotovoltaicos em geral apresentam baixa eficiência, variando de acordo com o material consegue-se apenas valores entre 7% e 18%. Assim, torna-se desafiador o desenvolvimento de tecnologias que visem aproveitar esta energia para o melhor uso da população. Dentro deste cenário, o presente trabalho tem por objetivo propor uma plataforma embarcada para controle do posicionamento realizado por motores elétricos, composta por software desktop, protocolo de comunicação e software embarcado. Através das informações definidas em software *desktop* e da integração entre plataforma proposta e um painel fotovoltaico, realiza-se o rastreamento solar baseando-se nas informações definidas pelo usuário e nos valores do calendário de tempo real do microcontrolador. Nos testes de integração realizados para a validação da plataforma, obteve-se resultados de acordo com o projetado durante a modelagem e implementação da plataforma.

**Palavras-chave:** Sistema Embarcado; Posicionamento de Motores; Energia Solar.

**DEVELOPMENT OF EMBEDDED PLATFORM FOR MOTOR POSITIONING WITH  
VALIDATION IN A PHOTOVOLTAIC SYSTEM**

**ABSTRACT**

Electric motors play a key role in modern society, being present in various sectors, such as industry, agriculture, academia, trade, power generation, among others. A limiting factor in the use of motors in Research and Development projects (R & D) is due to difficulties encountered in its configuration and control. There is a lack in technological solutions that provide the user with greater experience in the use of electric motors in the most diverse spheres. Among the many fields of knowledge that could make use of an intuitive system for positioning motors, stands the solar industry because of its availability and abundance. Photovoltaic systems usually have low efficiency varies according to the material can only values between 7% and 18%. Thus, it is challenging the development of technologies aimed at harnessing this energy to the best use of the population. Within this scenario, this paper aims to propose an embedded platform to control the positioning performed by electric motors, consisting of desktop software, communication protocol and embedded software. Through the information defined in desktop software and the integration of proposed platform and a PV panel, carried out the solar tracking relying on the user-defined information and real-time calendar values of the microcontroller. In integration testing for validation platform, results were obtained in accordance with the projected during the modeling and implementation of platform.

**Keywords: Embedded System; Motor Positioning; Solar energy;**

## LISTA DE FIGURAS

Figura 1: Etapas adotadas na metodologia.....	17
Figura 2: Estrutura Metálica do Protótipo de Seguidor Solar, Fonte (ALMEIDA, 2013) .....	19
Figura 3: Esquema de Conexão - Motores, Fonte: (ALMEIDA, 2013).....	20
Figura 4: Fluxo de sinais do sistema de controle, Fonte: (ALMEIDA, 2013).....	20
Figura 5: Ordens executadas pelo driver, Fonte: (ALMEIDA, 2013).....	21
Figura 6: Diagrama de Blocos do Sistema, Fonte(SOUZA, 2008).....	22
Figura 7: Diagrama completo do sistema experimental, Fonte: (FERREIRA, 2008).	23
Figura 8: Representação em Blocos das rotinas de cálculos, Fonte: (Ferreira,2008) .....	24
Figura 9: Software desktop, Fonte: (Teodoro, 2010).....	25
Figura 10: Diagrama em Blocos do Sistema, Fonte: (CÚNICO, 2006).....	26
Figura 11: Tela do Software de Controle, Fonte:(CÚNICO, 2006).....	27
Figura 12: Arquitetura de hardware do AstroFácil, Fonte:(SILVA, 2006).....	28
Figura 13: Comunicação entre os softwares.....	29
Figura 14: Estrutura do Pacote.....	30
Figura 15: Diagrama de Conexões da plataforma.....	35
Figura 16: Diagrama Controle em malha fechada.....	36
Figura 17: Protótipo do Painel fotovoltaico com uso de concentradores.....	37
Figura 18: Diagrama de conexões da plataforma proposta integrada ao sistema fotovoltaico.....	38
Figura 19: Kit STM32F4DISCOVERY.....	39
Figura 20: Modos de operação.....	40
Figura 21: Diagrama de atividades do modo de operação inicialização.....	41
Figura 22: Diagrama de atividades do modo de operação normal.....	42
Figura 23: Diagrama de atividades do modo de operação configuração.....	44
Figura 24: Diagrama de atividades do modo de operação standby.....	45
Figura 25: Diagrama de Casos de Uso.....	46
Figura 26: Tela Inicial Software Desktop.....	47
Figura 27: Painel de Controle.....	48
Figura 28: Sensores Ativados.....	49

Figura 29: Motor de Passo Modelo 28BYJ-48.....	50
Figura 30: Curva de Calibração e Equação Linear Aproximada.....	52
Figura 31: Kit de Testes da Plataforma.....	52
Figura 32: Configuração Software Desktop.....	53
Figura 33: Painel Fotovoltaico.....	54
Figura 34: Driver acionamento dz nano turbo Rossi.....	55
Figura 35: Tipo de Acionamento dz nano turbo.....	55
Figura 36: Integração do sistema com realimentação.....	56
Figura 37: Integração final do sistema.....	57
Figura 38: Configurações Software Desktop.....	57
Figura 39: Configuração do software desktop.....	58



## LISTA DE TABELAS

Tabela 1: Atividades Modo Normal.....	42
Tabela 2: Atividades Modo Configuração.....	43
Tabela 3: Atividades Modo Standby.....	44
Tabela 4: Tabela Ângulo-Tensão do Potenciômetro.....	51

## SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO.....	14
1.1 - Contexto.....	14
1.2 - Objetivo.....	16
1.3 - Metodologia.....	17
1.4 - Estrutura do Trabalho.....	18
CAPÍTULO 2 – TRABALHOS RELACIONADOS.....	19
CAPÍTULO 3 – PROTOCOLO DE COMUNICAÇÃO.....	29
3.1 – Estrutura do Pacote.....	30
3.2 - Comandos.....	30
3.2.1 – Enviar Amostras ADC Coletadas (Sentido – Embarcado - Desktop).....	30
3.2.2 – Enviar Data Calendário de Tempo Real (Sentido – Embarcado - Desktop) .....	30
3.2.3 – Entrar em Modo de Configuração (Sentido – Desktop – Embarcado).....	31
3.2.4 – Finalizar Modo de Configuração (Sentido – Desktop – Embarcado).....	31
3.2.5 – Envie-me Data Atual (Sentido – Desktop – Embarcado).....	31
3.2.6 – Restart (Sentido – Desktop – Embarcado).....	31
3.2.7 – Sensores Habilitados (Sentido – Desktop – Embarcado).....	31
3.2.8 – Enviar Posição Inicial (Sentido – Desktop – Embarcado).....	31
3.2.9 – Enviar Tempo de Atualização do Posicionamento (Sentido – Desktop – Embarcado).....	32
3.2.10 – Enviar Horário de Inicialização do Sistema (Sentido – Desktop – Embarcado).....	32
3.2.11 – Enviar Horário de Finalização do Sistema (Sentido – Desktop – Embarcado).....	32
3.2.12 – Envie-me Amostras ADC Coletadas (Sentido – Desktop – Embarcado). 32	
3.2.13 – Enviar Data do Software Desktop (Sentido – Desktop – Embarcado).....	32
3.2.14 – Enviar Passo em Ângulo (Sentido – Desktop – Embarcado).....	32
3.2.15 – Ativação por Pulso ou Nível (Sentido – Desktop – Embarcado).....	33
3.2.16 – Enviar Coeficientes da Função do Motor (Sentido – Desktop – Embarcado).....	33
3.2.17 – Enviar Posição Final (Sentido – Desktop – Embarcado).....	33
3.3 – Tabela de Informações.....	33

CAPÍTULO 4 – IMPLEMENTAÇÃO.....	35
4.1 Sistema Proposto.....	35
4.2 - Microcontrolador STM32F4DISCOVERY.....	38
4.3 – Software Embarcado.....	39
4.3.1 – Modos de Operação.....	39
4.3.2 – Modo de Operação Inicialização.....	40
4.3.3 – Modo de Operação Normal.....	41
4.3.4 – Modo de Operação Configuração.....	43
4.3.5 – Modo de Operação Standby.....	44
4.4 – Software Desktop.....	45
CAPÍTULO 5 – RESULTADOS E DISCUSSÕES.....	50
5.1 – Testes e Validação Utilizando Motores de Passo.....	50
5.2 – Validação com o Painel Fotovoltaico.....	53
CONCLUSÃO E PERSPECTIVAS FUTURAS.....	60
REFERÊNCIAS BIBLIOGRÁFICAS.....	61

# CAPÍTULO 1 - INTRODUÇÃO

## 1.1 - Contexto

Os motores elétricos exercem um papel fundamental na sociedade moderna, estando presente nos mais variados setores, tais como: a indústria, a agricultura, o ambiente acadêmico, o comércio e geração de energia, entre outros (SOUSA, 1995). Tal ideia pode ser comprovada pelo fato de que quase 50% de toda energia consumida no mundo são utilizadas por motores elétricos (SALERNO, 2000).

Um fator restritivo para uma maior utilização de motores elétricos na resolução de problemas, especialmente em projetos de Pesquisa & Desenvolvimento (P&D), ocorre devido às dificuldades encontradas na configuração e controle dos motores por pesquisadores de áreas não associadas às engenharias elétrica, eletrônica e mecânica.

Alguns autores elaboraram trabalhos no sentido de fornecer ao usuário bibliotecas ou plataformas específicas para a resolução de problemas que envolvem o controle de motores. As soluções mais utilizadas são técnicas de controle e automação, utilização de microcontroladores, criação de softwares *desktops* e utilização da ferramenta MATLAB. Algumas destas são encontradas em Moura (2010), Duany (2007), Almeida (2013), Cúnico (2006) e Ruzu (2006).

Nos trabalhos encontrados, notou-se que em todas as soluções adotadas as plataformas de controle e manipulação de motores elétricos estavam estritamente ligadas ao projeto específico que as originou, impossibilitando a reutilização da plataforma em novos projetos.

Portanto, há uma lacuna em soluções tecnológicas que proporcionem ao usuário uma maior experiência na configuração e utilização de motores elétricos nas mais diversificadas esferas acadêmicas.

Dentre as possíveis áreas do conhecimento que poderiam fazer uso de um sistema intuitivo de posicionamento de motores, o setor de energia solar merece uma atenção especial, pois, a fonte renovável solar é uma energia absolutamente pura, não poluente e que não produz produtos radioativos nem fumaça. Caracteriza-se então uma fonte ideal

para a geração de energia com foco na preservação do meio ambiente (SOUZA, 2008).

Segundo Souza (2008), há uma grande evidência de que a maior parte das tecnologias utilizadas não são ecologicamente adequadas e podem ocasionar graves e irreversíveis mudanças no clima do planeta, além disso, destaca-se que estas fontes energéticas em sua maioria não são renováveis e estão passíveis ao rápido esgotamento.

Tais fatores supracitados conduzem a um maior número de pesquisas com fontes que não poluam a atmosfera e não contribuam para o aumento da temperatura do planeta, dentre elas destacam-se energia eólica, biomassa e energia solar, pois, são abundantes, bem distribuídas e ecologicamente viáveis (SOUZA, 2008).

Dentre as fontes de energias renováveis, a energia solar oferece a maior parcela devido a sua disponibilidade e abundância. Segundo a Empresa de Pesquisa Energética, através do Plano Nacional de Energia 2030 do Ministério de Minas e Energia, a energia fotovoltaica é uma grande promessa para a geração distribuída (EPE, 2015).

A energia oriunda do sol pode ser utilizada para diversas finalidades, inclusive, substituir com vantagens formas convencionais de geração de energia, como por exemplo, Hidroelétrica, petróleo e gás natural (MME, 2015).

Mesmo com o progresso tecnológico e benefícios advindos da energia elétrica, cerca de um terço da população mundial ainda não tem acesso ao recurso (FERREIRA, 2008). O Brasil apresenta-se em melhor situação, porém, ainda ineficiente, pois, levando em consideração a grande extensão territorial do país e abundância de recursos energéticos, ainda existem regiões com problemas de suprimento energético.

De acordo com o MMA (Ministério do Meio Ambiente), a energia solar pode ser implementada com sucesso em qualquer latitude da extensão territorial do Brasil. Mesmo em regiões que apresentam índices de radiação reduzidos podem possuir um grande potencial de benefício energético (MME, 2015).

Um sistema fotovoltaico de forma geral, consegue gerar uma corrente CC quando ocorre uma junção semicondutora PN no material iniciada pela incidência de fótons em sua superfície. Enquanto a radiação solar estiver sobre a célula fotovoltaica será gerada energia (CARDOSO, 2012).

A intensidade da corrente elétrica gerada variará de acordo com a intensidade da luz. Uma célula fotovoltaica não armazena energia, apenas mantém o fluxo de elétrons de acordo com o campo elétrico enquanto houver a existência de incidência de luz. Para este efeito, denomina-se “Efeito Fotovoltaico” (NASCIMENTO, 2004).

Um limitante na utilização de sistemas fotovoltaicos para o aproveitamento da

energia proveniente do sol é a baixa eficiência das células. Variando de acordo com o material, consegue-se entre 7% e 18% de eficiência (NASCIMENTO, 2004).

Vale ressaltar que no contexto nacional, com o bom aproveitamento dessa tecnologia, o nosso problema energético poderia ser solucionado, visto que o Brasil possui dimensões continentais com grande incidência solar durante a maior parcela do ano. Assim, torna-se necessário o desenvolvimento de tecnologias que visem aproveitar esta energia para o melhor uso da população.

Pesquisas em desenvolvimento tentam buscar o aumento da eficiência de sistemas fotovoltaicos. Com diferentes enfoques os estudos tentam encontrar alternativas para a solução deste problema.

Desenvolvimento de novos materiais para construção de células, utilização de concentradores solares, rastreamento solar através da luminosidade, sistemas embarcados com uso de algoritmos para controle do Rastreamento do Ponto De Máxima Potência (MPPT – *Maximum Power Point Tracking*) e sistemas embarcados com implementação de Calendário de Tempo Real (RTC), são algumas das soluções adotadas até então para o aumento da eficiência em sistemas fotovoltaicos.

## 1.2 - Objetivo

Tendo em vista a importância dos motores em pesquisas e novos produtos nas mais diversas áreas do conhecimento, este trabalho tem como objetivo propor uma plataforma embarcada que possibilite a utilização de motores elétricos para controle de posicionamento, sem que o usuário possua conhecimento de programação de microcontroladores. A validação do sistema será realizada em um sistema fotovoltaico de geração de energia, onde será possível controlar o posicionamento de um sistema de rastreamento do sol, baseando-se em um relógio de tempo real.

De forma mais específica, neste trabalho pretende-se:

- Desenvolver um Software Embarcado responsável por desempenhar a função de *datalogger* e controle do sistema fotovoltaico através do motor;
- Desenvolver um Software Desktop capaz de configurar todos os parâmetros necessários para a utilização do motor e consequentemente do sistema

fotovoltaico, bem como, obter informações do sistema no decorrer do tempo através de sensores;

- Desenvolver um Protocolo de comunicação que possibilite a troca de informações entre o Software Embarcado-Desktop de forma eficiente;

### 1.3 - Metodologia

Para alcançar os objetivos deste trabalho as seguintes ações foram realizadas:

- Revisão bibliográfica sobre as principais soluções já adotadas no âmbito acadêmico;
- Estudo da plataforma ARM Cortex-M4;
- Estudo sobre o Desenvolvimento Desktop com linguagem de programação JAVA;
- Elaboração de documentos de requisitos, protocolo e especificação de projeto.

Os softwares embarcado-desktop são resultantes do estudo dos tópicos acima citados como pode ser visto na Figura 1 abaixo.

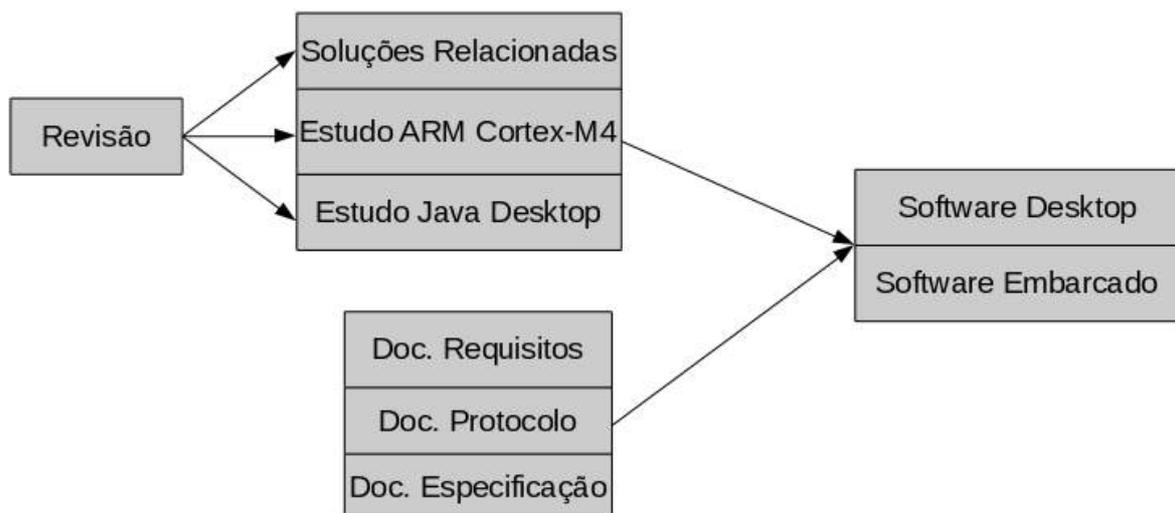


Figura 1: Etapas adotadas na metodologia

## 1.4 - Estrutura do Trabalho

Este trabalho está organizado da seguinte forma:

O Capítulo 2 aborda alguns trabalhos relacionados que apresentam relevância no contexto do projeto.

No Capítulo 3 apresenta-se o Protocolo de comunicação, bem como a organização dos seus dados.

O Capítulo 4 discorre sobre o Software embarcado-desktop, e suas características, tais como, módulos do software, modos de operação, interrupções e organização da plataforma.

O Capítulo 5 realiza a apresentação sobre os resultados obtidos com a integração da plataforma com o sistema fotovoltaico.

Por fim, trata-se das conclusões e sugestões futuras em função da análise dos resultados.

## CAPÍTULO 2 – TRABALHOS RELACIONADOS

Discorre-se nos parágrafos abaixo, de forma breve, a metodologia e resultados de algumas soluções já implementadas no âmbito acadêmico.

Almeida (2013) em seu trabalho, descreve o desenvolvimento de um protótipo de seguidor solar. Optou-se por um seguidor de um eixo horizontal, pois, além de ser uma opção de baixo custo e manutenção, é uma solução que se compatibiliza com o clima contextualizado no projeto, o nordeste do Brasil. Na Figura 2 abaixo pode-se visualizar a estrutura utilizada.



*Figura 2: Estrutura Metálica do Protótipo de Seguidor Solar, Fonte (ALMEIDA, 2013)*

Para o acionamento dos motores, Almeida (2013) utilizou um motor de passo em conjunto com uma caixa de redução. O esquema de conexão entre os componentes para o acionamento dos motores utilizados pelo autor pode ser visto na Figura 3 abaixo.

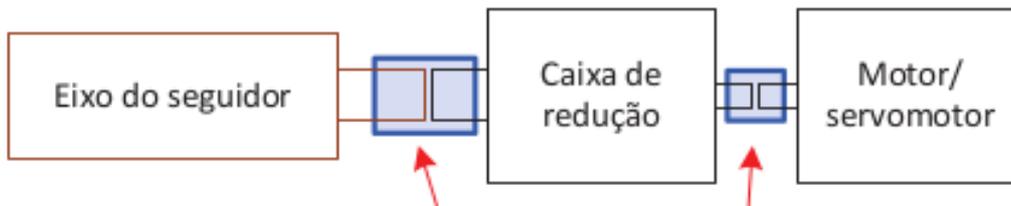


Figura 3: Esquema de Conexão - Motores, Fonte: (ALMEIDA, 2013)

Para o rastreamento do sol e controle do seguidor solar, Almeida (2013) utilizou sensor óptico, chaves de fim de curso e um *driver* programável para o controle do motor de passo. A Figura 4 abaixo mostra o fluxo de sinais no sistema de controle.

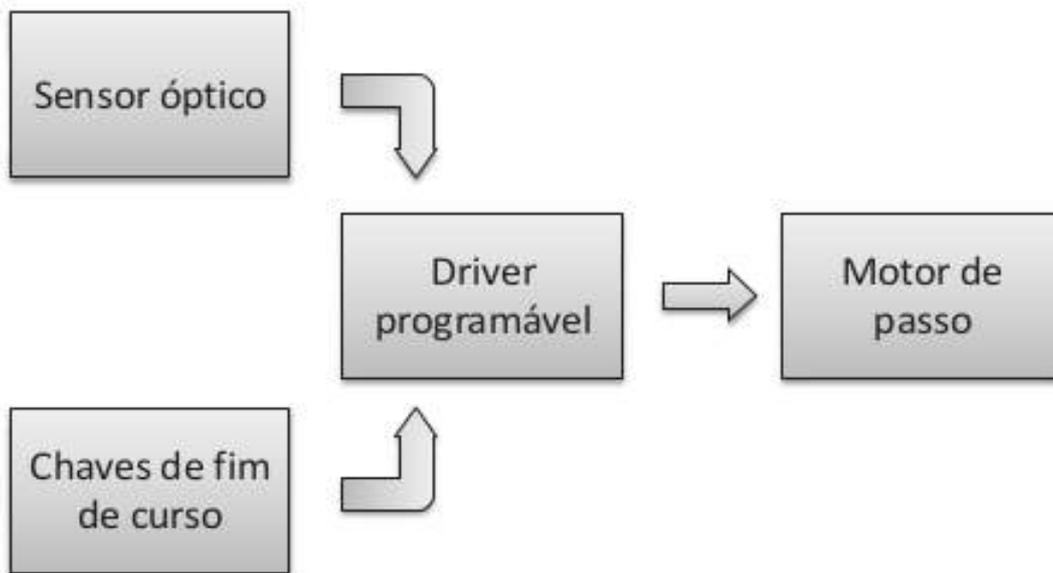


Figura 4: Fluxo de sinais do sistema de controle, Fonte: (ALMEIDA, 2013)

O sensor óptico e as chaves de fim de curso enviam sinais digitais para o driver, que, por conseguinte, controla o motor de passo. O *driver* programável foi configurado de acordo com a Figura 5 abaixo, onde, as duas entradas são o sensor óptico e as chaves.

<b>Input 1</b>	<b>Input 2</b>	<b>Movimento do motor</b>
1	1	Parado
0	1	Rotação horária
1	0	Rotação anti-horária

*Figura 5: Ordens executadas pelo driver, Fonte: (ALMEIDA, 2013)*

As medições realizadas por (ALMEIDA, 2013) durante um dia de operação, resultou em um ganho de quase 20% quando comparado ao sistema fixo. Por fim, Almeida (2013) propõe como melhorias, o emprego de outros tipos de controle em conjunto com a utilização de microcontroladores.

De forma semelhante a Almeida (2013), Souza (2008) desenvolveu a construção e montagem de um sistema de automatização para acompanhamento do movimento solar. O trabalho tem por objetivo utilizar a energia solar para aquecer e/ou vaporizar água para aplicações residenciais, comerciais e industriais.

Souza (2008) elaborou em seu trabalho um circuito específico para o controle do motor, que por sua vez não possui nenhum tipo de interação direta com o usuário, como, por exemplo, configuração de posicionamento e sensibilidade.

Para realizar o acompanhamento solar, Souza (2008) utilizou fotosensores, que, de acordo com a iluminação recebida aciona um circuito eletrônico, que, por sua vez, aciona um motor de corrente contínua. No final do dia, ou seja, de sua operação, a superfície atinge o ponto de fim de curso, acionando o seu retorno ao ponto inicial.

O diagrama de blocos do sistema de rastreamento e posicionamento do equipamento pode ser visto na Figura 6 abaixo.

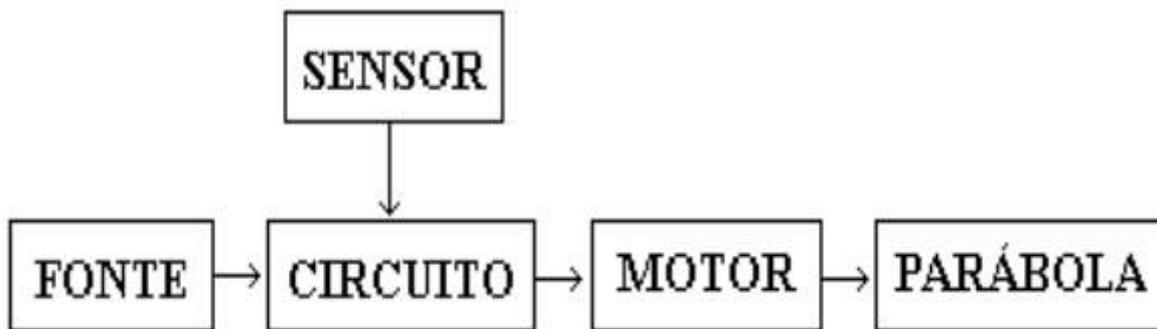


Figura 6: Diagrama de Blocos do Sistema, Fonte(SOUZA, 2008)

Souza (2008) obteve resultados satisfatórios com o sistema desenvolvido, conseguindo um aumento em torno de 14% na temperatura da água quando comparado ao sistema sem automação. O autor destaca que o sistema se torna viável a partir do momento que o retorno do capital investido ocorrerá em aproximadamente um ano de operação.

Com relação ao mecanismo de rastreamento solar, Souza (2008) propõe como possíveis melhorias, a utilização de motor de passo, pois, possibilita uma maior precisão. Sugere-se também a criação de um sistema eletrônico que permita a aquisição de dados em tempo real, com o objetivo de realizar um diagnóstico mais concreto do desempenho do sistema.

Segundo Ferreira (2008), sistemas de rastreamento solar com uso de sensores em geral é a solução mais simples, porém, estes sistemas apresentam em comum a limitação de funcionamento sob condições “não-previstas”, ocasionando em erros de rastreamento. Como exemplos destacam-se, ocorrência de nuvens ou sombras que encubram um ou todos os sensores.

Dentro deste contexto, Ferreira (2008) apresenta o desenvolvimento experimental de um conjunto de posicionador automático para módulos fotovoltaicos. A estrutura mecânica utilizada possibilita a movimentação da placa fotovoltaica em azimute e inclinação. No projeto, utilizaram-se motores de passo, controlador lógico programável (CLP), chaves de fim de curso e alguns componentes de baixo custo, sem usos de sensores no auxílio do posicionamento.

Na Figura 7 abaixo, pode-se visualizar o esquemático de todo o sistema desenvolvido.

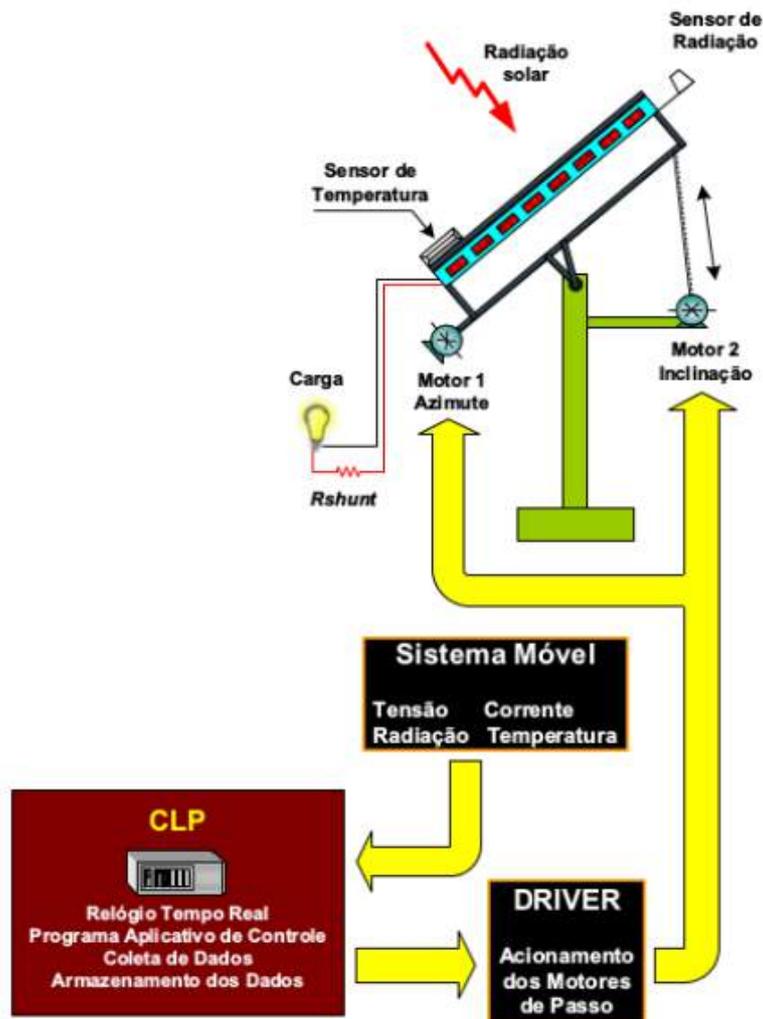


Figura 7: Diagrama completo do sistema experimental,  
 Fonte: (FERREIRA, 2008)

No sistema desenvolvido por Ferreira (2008), o CLP é responsável pelo controle do sistema móvel e também da coleta e armazenamento dos dados de alguns sensores. São armazenadas informações sobre tensão elétrica, corrente elétrica, temperatura e radiação solar. O sistema não envolve malha de realimentação, sendo calibrado uma vez ao dia, no início do movimento, através de uma chave de fim de curso.

O CLP executa o programa de controle que, a partir das equações de posicionamento do sol e de dados previamente informados, como posição geográfica e horário local, calcula a provável posição do sol e gera os sinais de controle para os motores de passo.

A interação entre usuário e sistema ocorre de forma engessada, pois, o seu controle é realizado apenas pelo programa executado no CLP. Desta forma, havendo a necessidade de alguma modificação nas configurações do sistema, necessita-se de um

prévio conhecimento da tecnologia, pois, deve-se modificar o código fonte e inseri-lo novamente na memória do CLP.

Para o cálculo dos ângulos (azimute e inclinação) e por conseguinte o correto posicionamento do sistema, Ferreira (2008) utilizou como base o Calendário de Tempo Real (RTC) implementado pelo CLP e equações para determinar o horário solar. A partir destas informações, as rotinas de conversão de parâmetros puderam ser implementadas. O diagrama de blocos com as rotinas de cálculos pode ser visto abaixo na Figura 8.

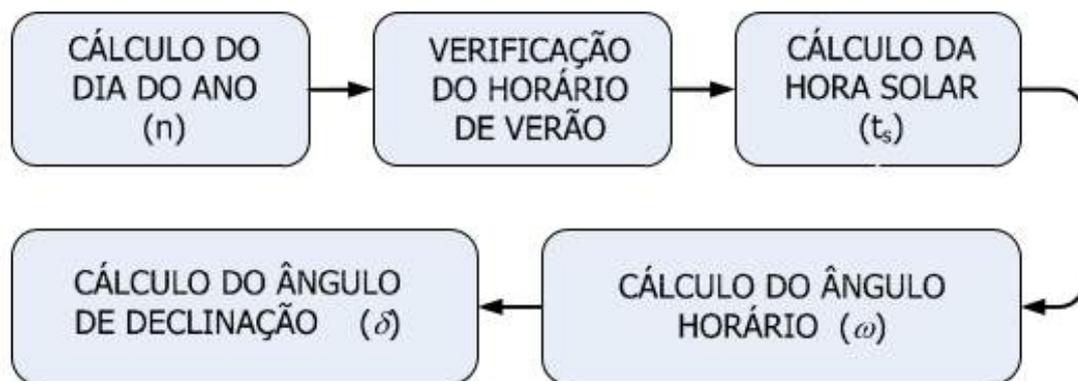


Figura 8: Representação em Blocos das rotinas de cálculos, Fonte: (Ferreira,2008)

Os cálculos inseridos no diagrama de blocos são realizados diariamente pelo programa de controle executado no CLP.

Os dados do RTC utilizados como base no cálculo dos ângulos, são implementados internamente pelo CLP, e fornece as seguintes características:

- Variável inteira no formato DDMM;
- Variável inteira no formato HHMM.

As medições realizadas por Ferreira (2008) ao longo de um dia de operação do sistema, demonstraram um ganho de 40% na geração de energia, quando comparado ao mesmo sistema fixo. As 09h31min a diferença entre os dois sistemas atingiu a máxima de 150% de diferença.

Realizando comparação de custos na aquisição dos sistemas fixo e móvel, verificou-se um custo 35,7% superior para o sistema fotovoltaico móvel. Entretanto, esse custo poderá ser recompensado pelo ganho médio de 40% na geração de energia.

Em conclusão, Ferreira (2008) destaca a viabilidade do sistema do ponto de vista econômico e operacional. O autor ainda destaca como continuidade do trabalho a

construção/aquisição de um controlador dedicado, pois, o CLP devido aos seus custos, dimensões físicas e consumo de energia não é a melhor opção.

De forma semelhante a Ferreira (2008), Teodoro (2010) desenvolveu em seu trabalho um sistema de rastreamento solar utilizando um algoritmo que calcula a trajetória do sol a cada instante do ano. O algoritmo utilizado por Teodoro (2010) pode ser encontrado em Stine (2001).

Na abordagem realizada por Teodoro (2010), optou-se por um sistema sem microcontroladores ou sensores para o controle de posicionamento. Desta forma, o controle do sistema fotovoltaico é realizado por um software desktop, conectado ao sistema eletrônico de controle do motor e aquisição de dados através da porta paralela.

Para a aquisição e configuração de algumas informações do sistema, Teodoro (2010) desenvolveu um software *desktop*. No software desktop o usuário consegue: editar a latitude do local e o dia do ano, visualizar algumas informações acerca do sistema fotovoltaico, como, voltagem, ângulo e posição do motor. O software pode ser visto na Figura 9 abaixo.

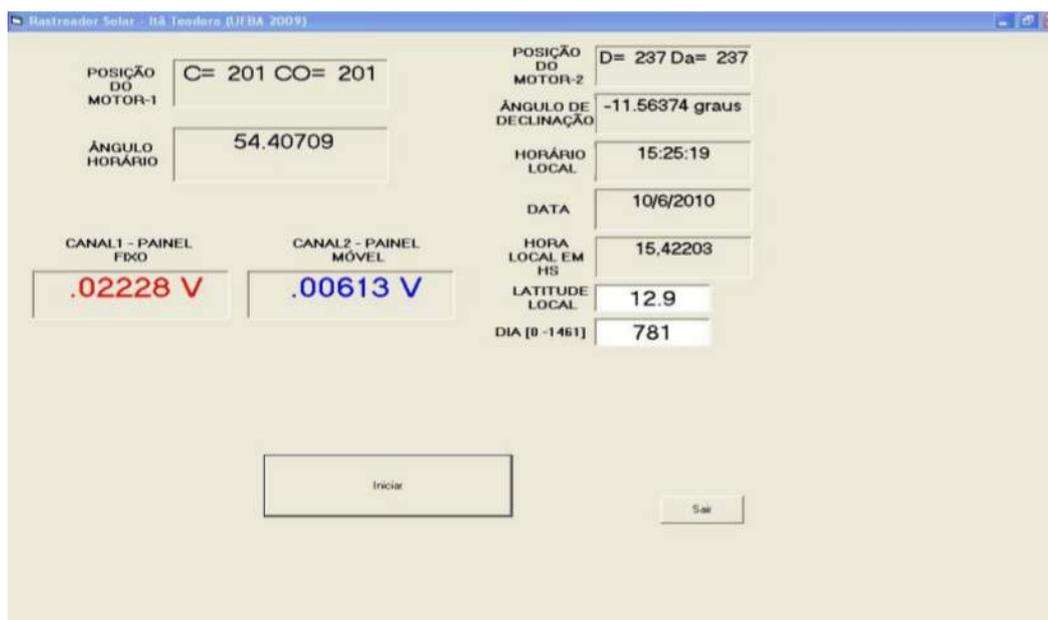


Figura 9: Software desktop, Fonte: (Teodoro, 2010)

Como pode ser visto na Figura 9 acima, o software não possibilita editar informações referentes ao motor, como por exemplo, tipo do motor ou tipo de acionamento do motor. Trata-se então de um software específico para a eletrônica e o motor escolhido no trabalho desenvolvido por Teodoro (2010).

O sistema de rastreamento proposto por Teodoro (2010) obteve êxito em testes

realizados durante 15 dias. Com o correto posicionamento durante os dias de operação, mediu-se um ganho de aproximadamente 39% em relação ao painel fixo.

Outra possível solução para o aumento da eficiência em sistemas fotovoltaicos é a otimização do ponto de operação de máxima potência. Para cada valor de irradiação solar existe apenas um ponto onde o painel solar entrega a máxima potência para a carga, esse valor é o Ponto de Máxima Potência (MPP) (CARVALHO, 2012). Nessa temática, diversos estudos estão sendo realizados. Magalhães (2006), realizou estudos com o objetivo de aplicar técnicas de linearização e alguns princípios de controle ótimo para modelar um sistema de rastreamento de potência máxima no qual um gerador solar oferece para uma bateria.

Carvalho (2012) apresenta uma abordagem de rastreamento do ponto de máxima potência com o método de “perturbação” e “observação”, de modo que, apenas com a corrente de saída do painel é mantida a perfeita busca do MPP. O algoritmo desenvolvido pelo autor, além de oferecer simples implementação, demonstrou rápida convergência nos resultados em simulações.

Cardoso (2012) descreve em seu trabalho a implementação de um sistema de rastreamento solar com microcontrolador utilizando controle de ponto de máxima potência. Para desenvolver o sistema o autor utilizou sensores associado ao algoritmo de rastreamento do ponto de máxima potência. O algoritmo escolhido para a implementação foi proposto por Carvalho (2012).

Em um âmbito geral, alguns sistemas embarcados para controle de motores e consequentemente do equipamento final foram propostos. Cúnico (2006) desenvolveu um sistema embarcado para o posicionamento de antenas. O controle é realizado com o acionamento de dois motores de passo que controla o azimute e inclinação da antena. O sistema está disposto de acordo com a Figura 10 abaixo.



*Figura 10: Diagrama em Blocos do Sistema, Fonte: (CÚNICO, 2006)*

O sistema proposto por Cúnico (2006) apresenta diversas vantagens em relação aos sistemas mostrados anteriormente, pois, oferece ao usuário fácil configuração e

manipulação dos parâmetros do sistema através do software desktop mostrado na Figura 11. Ainda assim, a arquitetura do sistema não tem como caráter a universalidade, pois, foi desenvolvida de modo a integrar-se somente ao sistema proposto.



Figura 11: Tela do Software de Controle, Fonte: (CÚNICO, 2006)

Silva (2006) apresentou em seu trabalho um sistema embarcado denominado AstroFácil, a ser utilizado na automação de telescópios de pequeno porte visando facilitar a operação dos equipamentos por amadores.

Silva (2006) desenvolveu um manete, que, de forma simplificada é um dispositivo embarcado que possui teclado para entrada de dados e *display* para apresentação das informações. Utilizando dois microcontroladores e implementando um RTC, chegou-se a arquitetura do Sistema embarcado que pode ser visto na Figura 12 abaixo.

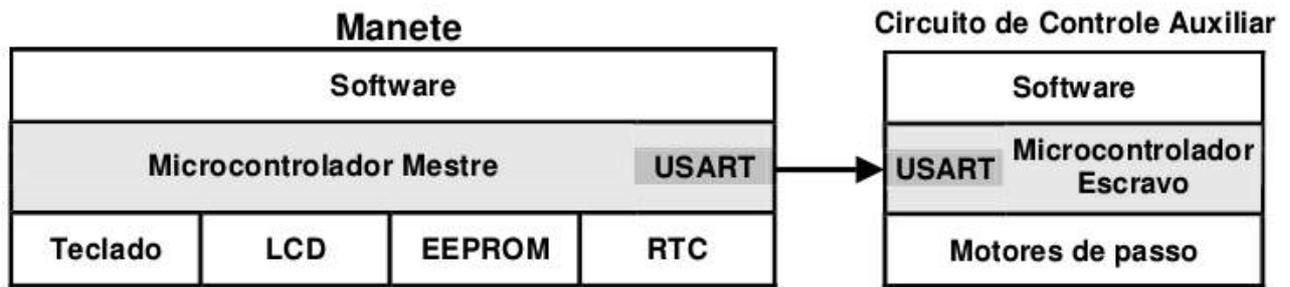


Figura 12: Arquitetura de hardware do AstroFácil, Fonte: (SILVA, 2006)

O manete produzido por Silva (2006) realizam as funções de, armazenar informações de corpos celestes, posicionar o telescópio através das coordenadas ou nome do astro, acompanhar o astro de acordo com seu movimento aparente e guiar o telescópio livremente.

Além do sistema proposto por Silva (2006) obter sucesso em simulações, obteve-se êxito também em testes iniciais com foco na integração do sistema. Propõe-se como próxima etapa a validação completa do sistema.

### CAPÍTULO 3 – PROTOCOLO DE COMUNICAÇÃO

O trabalho proposto tem em seus objetivos o desenvolvimento de software embarcado e software *desktop*. A partir do software *desktop* pode-se configurar os parâmetros necessários para o funcionamento do sistema embarcado. Horário inicial e final de operação do sistema, controle dos sensores ativados, ângulo inicial de operação, configuração da data do RTC no software embarcado, são algumas dentre as várias funcionalidades fornecidas pelo software *desktop*.

Para prover todas as funcionalidades do software *desktop* de forma satisfatória, implementou-se um protocolo de comunicação específico para o sistema proposto. Desta forma houve uma padronização nas mensagens trocadas entre o software *desktop* e embarcado. Na Figura 13 pode-se visualizar a forma de comunicação entre os softwares.



Figura 13: Comunicação entre os softwares

Em ambos os sistemas existe um módulo específico para realizar o tratamento do pacote recebido e encaminhá-lo corretamente para a sua rotina de execução.

Os dados são transferidos por comunicação serial, que possui taxa de 11500 Bauds/s. Como o tamanho total do pacote a ser transferido é de 11 bytes, utilizando comunicação serial consegue-se transmitir aproximadamente 1 pacote a cada milissegundo, valor totalmente viável no contexto do sistema proposto, pois, não impacta na experiência do usuário com o software.

### 3.1 – Estrutura do Pacote

O pacote do protocolo implementado é formado por 4 campos, que totalizam 11 Bytes. A Figura 14 abaixo o apresenta.

<b>Endereço</b>	<b>Tipo de Comando</b>	<b>Dados</b>	<b>Fim</b>
<b>1 Byte</b>	<b>1 Byte</b>	<b>8 Bytes</b>	<b>1 Byte</b>

Figura 14: Estrutura do Pacote

Segue abaixo a definição para cada campo do pacote.

- **Endereço:** Define o endereço do microcontrolador emissor/receptor da mensagem;
- **Tipo de Comando:** Especifica o comando acionado pelo *desktop* ou microcontrolador;
- **Dados:** Informação transferida entre os dispositivos;
- **Fim:** Indicador de fim de pacote;

### 3.2 - Comandos

Os comandos abaixo são implementados no protocolo, com diferenciação quanto ao seu sentido que pode ser *Embarcado-Desktop* ou *Desktop-Embarcado*.

#### 3.2.1 – Enviar Amostras ADC Coletadas (Sentido – Embarcado - *Desktop*)

Informação referente ao valor lido no ADC e calendário de tempo real são transmitidas. Neste comando o campo “dados” tem o seguinte formato: (Ano/Mês/Dia/Horas/Minutos/Segundos/Sensor/Valor) .

#### 3.2.2 – Enviar Data Calendário de Tempo Real (Sentido – Embarcado - *Desktop*)

Comando que envia ao software *desktop* a data armazenada no calendário de tempo real.

### **3.2.3 – Entrar em Modo de Configuração (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado a intenção de iniciar o seu o modo de configuração.

### **3.2.4 – Finalizar Modo de Configuração (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado a intenção finalizar o modo de configuração.

### **3.2.5 – Envie-me Data Atual (Sentido – *Desktop* – Embarcado)**

Comando que requisita a data armazenada no software Embarcado.

### **3.2.6 – *Restart* (Sentido – *Desktop* – Embarcado)**

Comando que indica a intenção de reiniciar o software embarcado.

### **3.2.7 – Sensores Habilitados (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado quais dos sensores estarão habilitados.

### **3.2.8 – Enviar Posição Inicial (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado a posição (ângulo) diária inicial do equipamento.

### **3.2.9 – Enviar Tempo de Atualização do Posicionamento (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado o intervalo de tempo requerido para a mudança de posicionamento do equipamento.

### **3.2.10 – Enviar Horário de Inicialização do Sistema (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado o horário para a inicialização do sistema (Posicionamento, Coletas).

### **3.2.11 – Enviar Horário de Finalização do Sistema (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado o horário para a finalização do sistema (Posicionamento, Coletas).

### **3.2.12 – Envie-me Amostras ADC Coletadas (Sentido – *Desktop* – Embarcado)**

Comando que requisita as amostras coletadas que estão armazenada no software Embarcado.

### **3.2.13 – Enviar Data do Software Desktop (Sentido – *Desktop* – Embarcado)**

Comando que envia o horário atual do *Desktop* para o software embarcado. Após a recepção o software embarcado redefinirá seu RTC para a data recebida. Neste comando o campo “dados” tem o seguinte formato: (Ano/Mês/Dia/Horas/Minutos/Segundos).

### **3.2.14 – Enviar Passo em Ângulo (Sentido – *Desktop* – Embarcado)**

Comando que indica ao software embarcado o ângulo que será acrescido a cada passo.

### 3.2.15 – Ativação por Pulso ou Nível (Sentido – *Desktop* – Embarcado)

Comando que indica ao software embarcado o tipo de ativação do motor.

### 3.2.16 – Enviar Coeficientes da Função do Motor (Sentido – *Desktop* – Embarcado)

Comando que indica ao software embarcado os coeficientes (A, B) que compõem a função de aproximação de cada motor. A partir dos valores (A, B), para cada voltagem será possível saber a posição do sistema.

### 3.2.17 – Enviar Posição Final (Sentido – *Desktop* – Embarcado)

Comando que indica ao software embarcado a posição (ângulo) diária final do equipamento.

## 3.3 – Tabela de Informações

Nome do Comando	Formato do Campo “Comando”
Enviar Amostra ADC (Sentido – Embarcado - Desktop)	00000001
Enviar Data Calendário de Tempo Real (Sentido – Embarcado - Desktop)	00000010
Entrar em Modo de Configuração (Sentido – <i>Desktop</i> – Embarcado)	00000011
Finalizar Modo de Configuração (Sentido – <i>Desktop</i> – Embarcado)	00000100
Envie-me Data Atual (Sentido – <i>Desktop</i> – Embarcado)	00000101
Restart (Sentido – <i>Desktop</i> – Embarcado)	00000110
Sensores Habilitados (Sentido – <i>Desktop</i> – Embarcado)	00000111

Enviar Posição Inicial (Sentido – <i>Desktop</i> – Embarcado)	00001000
Enviar Tempo de Atualização do Posicionamento (Sentido – <i>Desktop</i> – Embarcado)	00001001
Enviar Horário de Inicialização do Sistema (Sentido – <i>Desktop</i> – Embarcado)	00001010
Enviar Horário de Finalização do Sistema (Sentido – <i>Desktop</i> – Embarcado)	00001011
Envie-me Amostras ADC Coletadas (Sentido – <i>Desktop</i> – Embarcado)	00001100
Enviar Data do Software Desktop (Sentido – <i>Desktop</i> – Embarcado)	00001101
Enviar Passo em Ângulo (Sentido – <i>Desktop</i> – Embarcado)	00001111
Ativação por Pulso ou Nível (Sentido – Desktop – Embarcado)	00010000
Enviar Coeficientes da Função do Motor (Sentido – Desktop – Embarcado)	00010001
Enviar Posição Final (Sentido – <i>Desktop</i> – Embarcado)	00010010

O protocolo de comunicação implementado possui fundamental importância na plataforma proposta, pois, além de padronizar as trocas de informações entre os softwares *desktop* e embarcado, permite realizar configurar e controlar o posicionamento do motor completamente. Ressalta-se também que o protocolo oferece a possibilidade de expansão da aplicação do software *desktop*, que possibilitará inclusão de novos comandos.

## CAPÍTULO 4 – IMPLEMENTAÇÃO

Como forma de organização, e para o melhor entendimento, o Capítulo 4 está dividido de acordo com o desenvolvimento dos dois Softwares, embarcado e *desktop*.

No primeiro momento discorre-se sobre o microcontrolador utilizado e a arquitetura do software embarcado, envolvendo, os modos de operação, as atividades realizadas em *background* e *foreground*, interrupções, e diagramas.

O Capítulo estende-se com a apresentação do software *desktop*, abordando as suas funcionalidades e interface com o usuário.

### 4.1 Sistema Proposto

O sistema proposto nesse trabalho, tem como objetivo o desenvolvimento de uma plataforma embarcada que possibilite a utilização de motores elétricos. O caráter abrangente da plataforma possibilita ao usuário a possibilidade de integrá-lo de acordo com a necessidade do seu sistema específico. Na Figura 15 abaixo, pode-se visualizar o diagrama de conexões entre os componentes da plataforma.

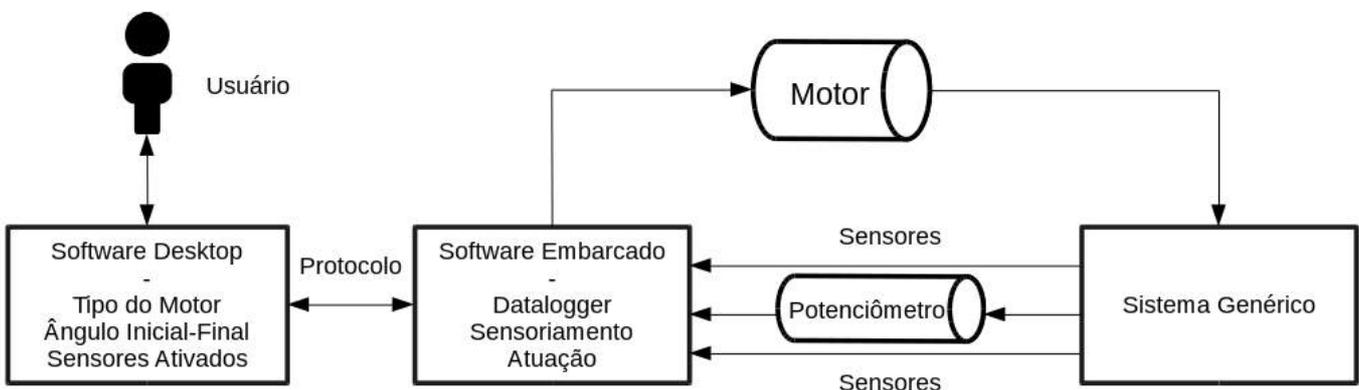


Figura 15: Diagrama de Conexões da plataforma

Em um primeiro estágio de interação, o usuário configura o software *desktop* indicando os parâmetros necessários para o funcionamento do motor e sistema. As informações são

enviadas para o microcontrolador via interface serial de acordo com o estabelecido no protocolo de comunicação.

Com o software embarcado já configurado, tem-se então, autonomia para realizar o controle do motor. Este controle é realizado através de um sistema de realimentação em malha fechada que pode ser visto na Figura 16 abaixo.

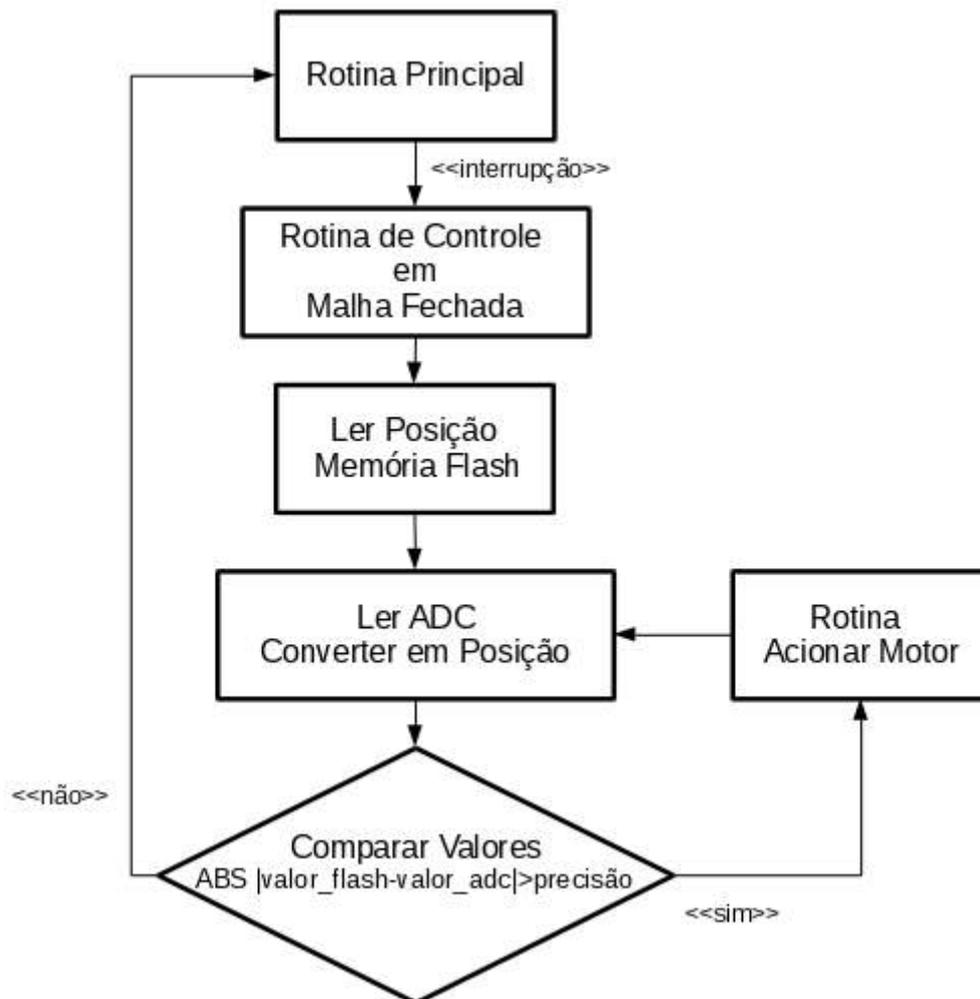


Figura 16: Diagrama Controle em malha fechada

O potenciômetro linear oferece ao software embarcado o real posicionamento do sistema, este valor obtido através do conversor analógico digital (ADC) é então comparado ao posicionamento armazenado na memória *flash* e previsto para o horário atual de acordo com o RTC do microcontrolador. A partir da diferença entre os valores ocorre a decisão entre o acionamento do motor ou a manutenção da sua inércia.

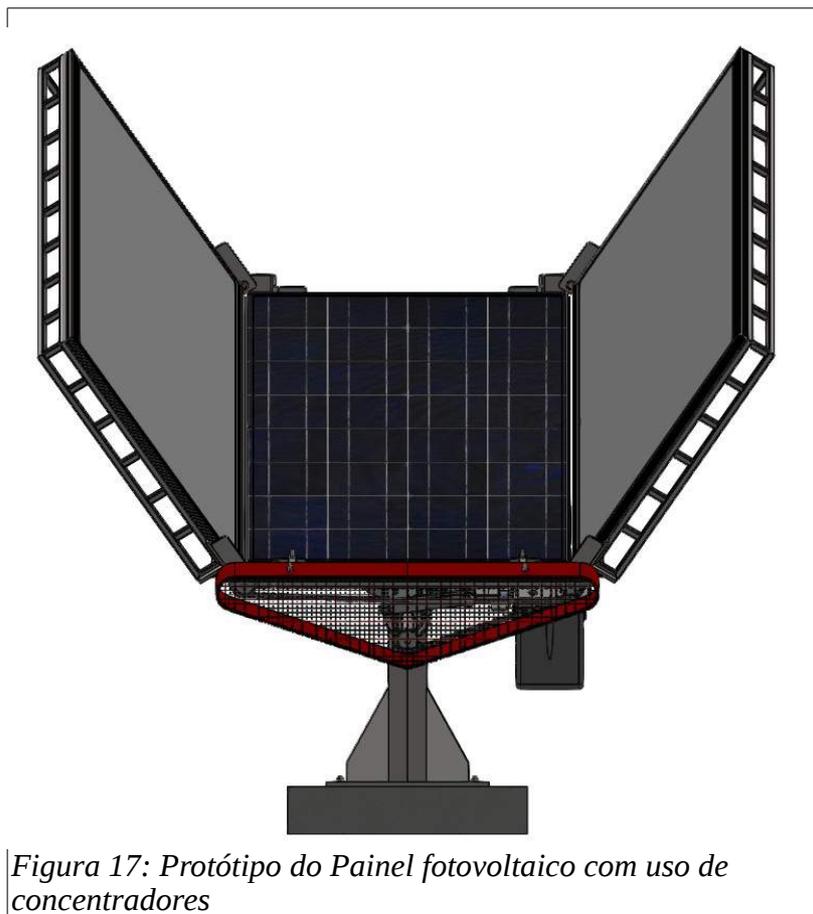
Um potenciômetro (ou mecanismo similar) desempenha papel fundamental no

sistema de realimentação em malha fechada. Como para cada aplicação da plataforma pode-se utilizar potenciômetros diferentes, faz-se necessário a criação de mecanismos que torne a plataforma independente do tipo de potenciômetro. Isto é realizado através dos valores dos coeficientes angular e linear da curva de calibração do potenciômetro.

Os valores dos coeficientes angular e linear da curva de calibração do indicados como parâmetros no software *desktop*, a partir destas informações é possível fazer a conversão voltagem-posicionamento de forma transparente.

Ressalta-se que para a integração entre um sistema controlado por um motor elétrico e a plataforma, alguns requisitos básicos devem ser obedecidos, são eles: potenciômetro linear acoplado ao sistema, motor com modo de ativação suportado pela plataforma, tensão máxima fornecida pelos sensores do sistema genérico de 3 Volts.

Como mencionado no Capítulo 1, a plataforma proposta será validada através de um sistema de geração de energia fotovoltaica com uso de concentradores. A estrutura mecânica do sistema pode ser vista na Figura 17 abaixo.



A movimentação dos concentradores será realizada pelo motor elétrico modelo dz nano turbo, este por sua vez está conectado e gerenciado pelo software embarcado. O sistema fotovoltaico integrado a plataforma proposta resulta no diagrama da Figura 18 abaixo.

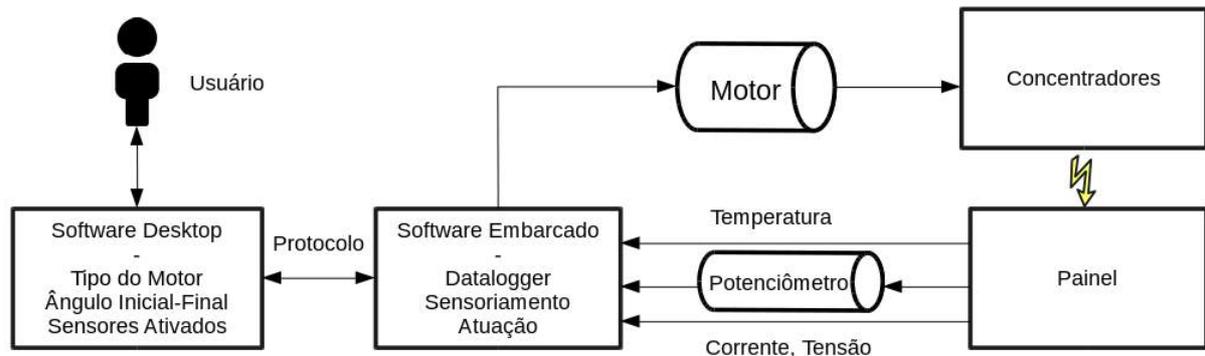


Figura 18: Diagrama de conexões da plataforma proposta integrada ao sistema fotovoltaico

## 4.2 - Microcontrolador STM32F4DISCOVERY

A plataforma escolhida para a implementação do sistema embarcado foi o kit STM32F4Discovery composto pelo microcontrolador STM32F407VGT6, pois, entre os kits disponíveis no mercado e na instituição este microcontrolador foi o que melhor se adequou as necessidades do sistema proposto. Dentre as suas características destacam-se:

- Núcleo ARM 32 bits Cortex-M4;
- Memória Flash 1 Mb;
- Clock de até 168 Mhz;
- Até 24 canais de entradas analógicas em 3 ADC'S de aquisição simultânea;
- Resolução de 12 bits do conversor analógico digital;
- Implementação nativa de RTC;
- Simulação de Comunicação Serial via USB On The Go;
- Possibilidade de programação utilizando somente software livre;

O Kit STM32F4DISCOVERY utilizado, pode ser visto na Figura 19 abaixo.



Figura 19: Kit STM32F4DISCOVERY

O sistema *host* utilizado para o desenvolvimento foi um Notebook HP G42 com a distribuição Fedora 21 KDE. Para a construção do software embarcado, que vai desde escrita, depuração e compilação do código fonte até a gravação no microcontrolador, foram utilizadas bibliotecas gratuitas.

Adotou-se ao projeto o OpenOCD (2015) e Telnet com a finalidade de depurar, programar e realizar a gravação do código no microcontrolador. A escrita do código foi realizada na linguagem de programação C utilizando o ambiente de desenvolvimento Eclipse (2015).

### 4.3 – Software Embarcado

Esta seção apresenta a arquitetura do software embarcado desenvolvido, explanando os modos de operação do sistema, os diagramas de estados do software, as interrupções e as suas prioridades.

#### 4.3.1 – Modos de Operação

O software embarcado pode realizar a sua operação diante de alguns contextos. Longos

intervalos de tempo sem atuação, momentos onde ocorrem ajustes de suas configurações, período durante a inicialização do software são alguns dos contextos encontrados e previstos na programação do seu software.

Denomina-se como modo de operação o comportamento e a maneira de executar ações seguindo os mesmos passos em uma determinada situação. Neste sentido, previu-se que o software embarcado proposto necessitaria de 4 modos de operação para compreender todo o seu universo de atuação. Os modos de operação podem ser visualizados na Figura abaixo.

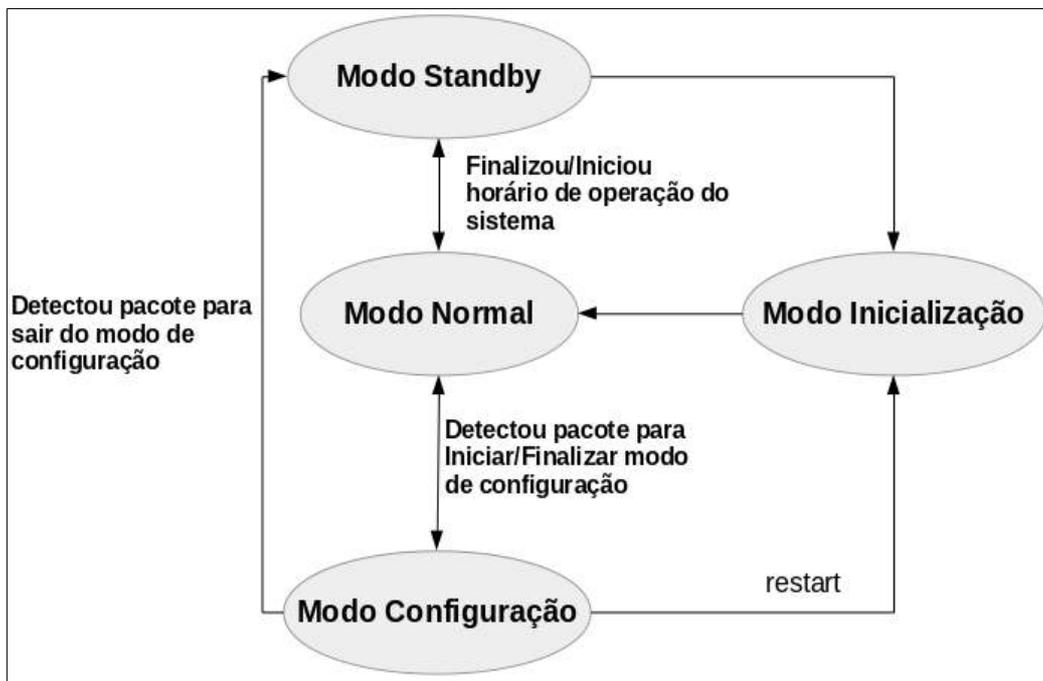


Figura 20: Modos de operação

#### 4.3.2 – Modo de Operação Inicialização

Após o sistema ser iniciado, este modo de operação invariavelmente será o primeiro a ser executado. Nesta etapa são realizadas verificações em alguns periféricos do microcontrolador e algumas configurações, tais como, definição *clock*, inicialização dos temporizadores, habilitação das interrupções, ajuste dos conversores A/D. O diagrama com as atividades do modo de operação inicialização pode ser visto na Figura 21 abaixo.

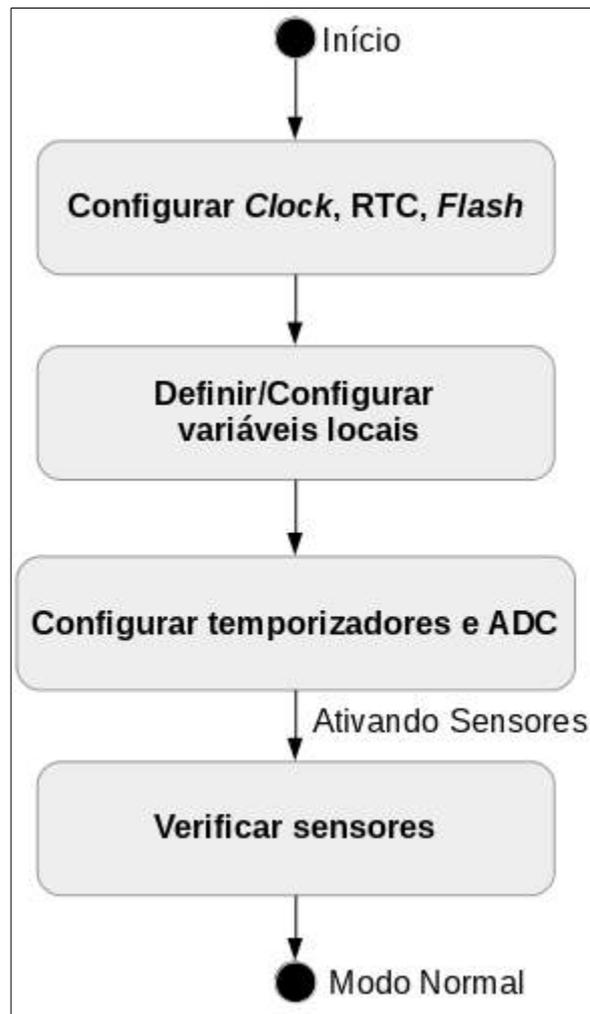


Figura 21: Diagrama de atividades do modo de operação inicialização

### 4.3.3 – Modo de Operação Normal

Nessa etapa são feitas as principais atividades do software embarcado. Inicia-se e finaliza-se este modo de operação de forma automática, de acordo com o horário do RTC, ou seja, durante o horário de operação indicado no software *desktop*.

No modo normal, o sistema deverá ser capaz de liberar o processamento para as interrupções, são elas: interrupção para controle em malha fechada e posicionamento do motor, interrupção para leitura dos ADC's, interrupção para a entrada em modo de configuração.

Segue abaixo na Tabela 1 as atividades a serem executadas nesse estado, elas

estão diferenciadas por execução em *Background* e execução em *Foreground*.

Background	Foreground
Manter calendário de tempo real	Interrupção para controle em malha fechada e posicionamento do motor
Detecção de preâmbulo	Interrupção para atualização dos ADC's
Interpretação de pacotes <i>recebidos</i>	Interrupção para entrada em modo de configuração.
Inserir informações coletadas dos sensores na memória <i>flash</i>	

Tabela 1: Atividades Modo Normal

As ações em *background* podem liberar o processamento para as interrupções definidas como tarefas de *foreground*, e após o processamento da interrupção retorna-se a tarefa em *background*. Na Figura abaixo é possível visualizar o diagrama com as atividades do modo de operação normal.

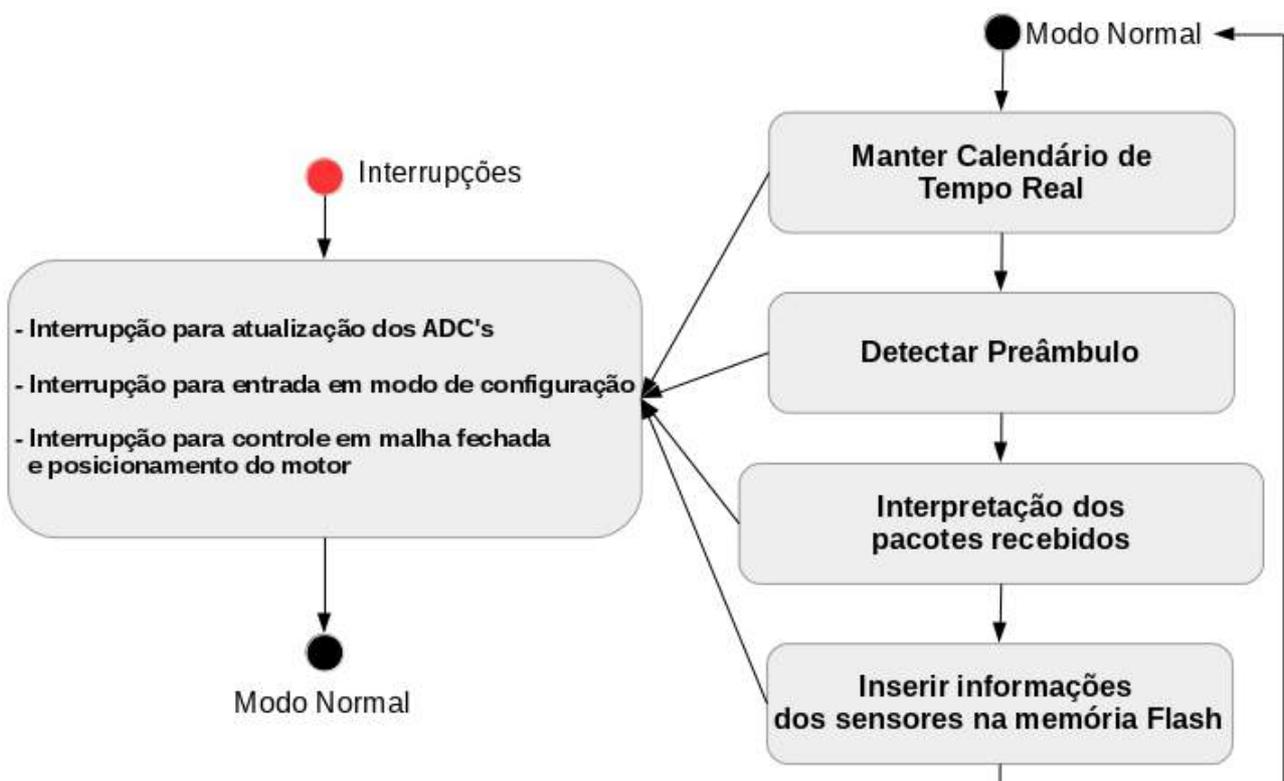


Figura 22: Diagrama de atividades do modo de operação normal

Nenhum tipo de modificação nas configurações (sensores ativados, ângulo inicial,

horário de operação, entre outros) ocorre durante este modo de operação. Desta forma, havendo a necessidade de mudança, inicia-se a interrupção responsável por ativar o modo de operação configuração.

#### 4.3.4 – Modo de Operação Configuração

Nessa etapa são feitas configurações acerca das instruções enviadas pelo software *desktop*. Chega-se a este modo de operação através da detecção do pacote específico para a ativação do modo de configuração.

No modo configuração, o sistema deverá liberar o processamento para as interrupções, são elas: Interrupção do RTC, interrupção para execução dos comandos recebidos.

Segue abaixo na Tabela 2 as atividades a serem executadas nesse estado, elas estão diferenciadas por execução em *Background* e execução em *Foreground*.

<b>Background</b>	<b>Foreground</b>
Manter calendário de tempo real	Interrupção para execução dos pacotes recebidos
Detecção de preâmbulo	
Interpretação de pacotes <i>recebidos</i>	

*Tabela 2: Atividades Modo Configuração*

As ações que podem ser realizadas com a interrupção para execução dos pacotes recebidos, foram listadas no Capítulo 3, seção 3.3 – Tabela de informações. Todas as mensagens do sentido *desktop*-embarcado podem ser executadas na interrupção.

Na Figura 23 abaixo pode-se visualizar o diagrama com as atividades do modo de operação configuração.

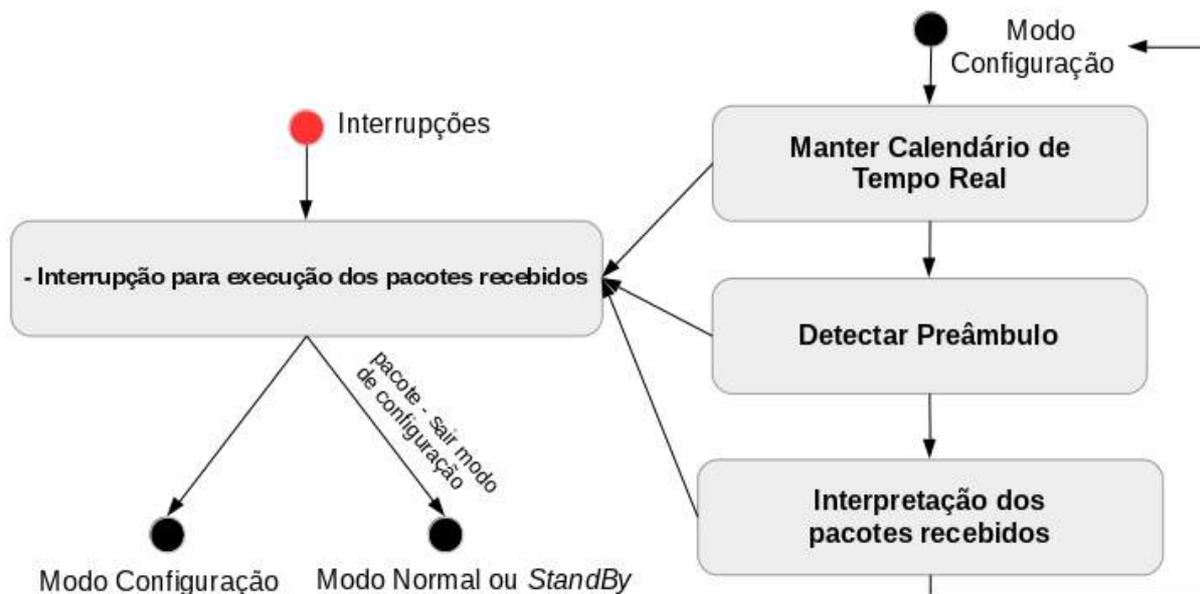


Figura 23: Diagrama de atividades do modo de operação configuração

Ao receber o pacote que indica a finalização do modo de configuração, o software embarcado retorna ao modo de operação anterior, modo normal ou standby.

#### 4.3.5 – Modo de Operação Standby

Este modo de operação é habilitado/desabilitado, assim como o modo normal, através de um horário previamente escolhido pelo usuário e regido pelo calendário de tempo real. Este modo de operação tem como principal objetivo realizar a menor quantidade de processamento possível, de forma há economizar energia. No modo *Standby*, o sistema será capaz de manter o calendário de tempo real e interpretar os pacotes recebidos.

Na Tabela 3 abaixo, pode-se visualizar as ações realizadas neste modo de operação.

Background	Foreground
Manter calendário de tempo real	Interrupção para entrada em modo de configuração.
Detecção de preâmbulo	
Interpretação de pacotes <i>recebidos</i>	

Tabela 3: Atividades Modo Standby

Este modo será acionado automaticamente na ocorrência da finalização do horário de operação do software embarcado, representados pelos pacotes de horário inicial de operação e horário final de operação. De forma análoga ocorre o retorno para o modo de operação normal. Na Figura a seguir é possível visualizar o diagrama com as atividades do modo de operação *standby*.

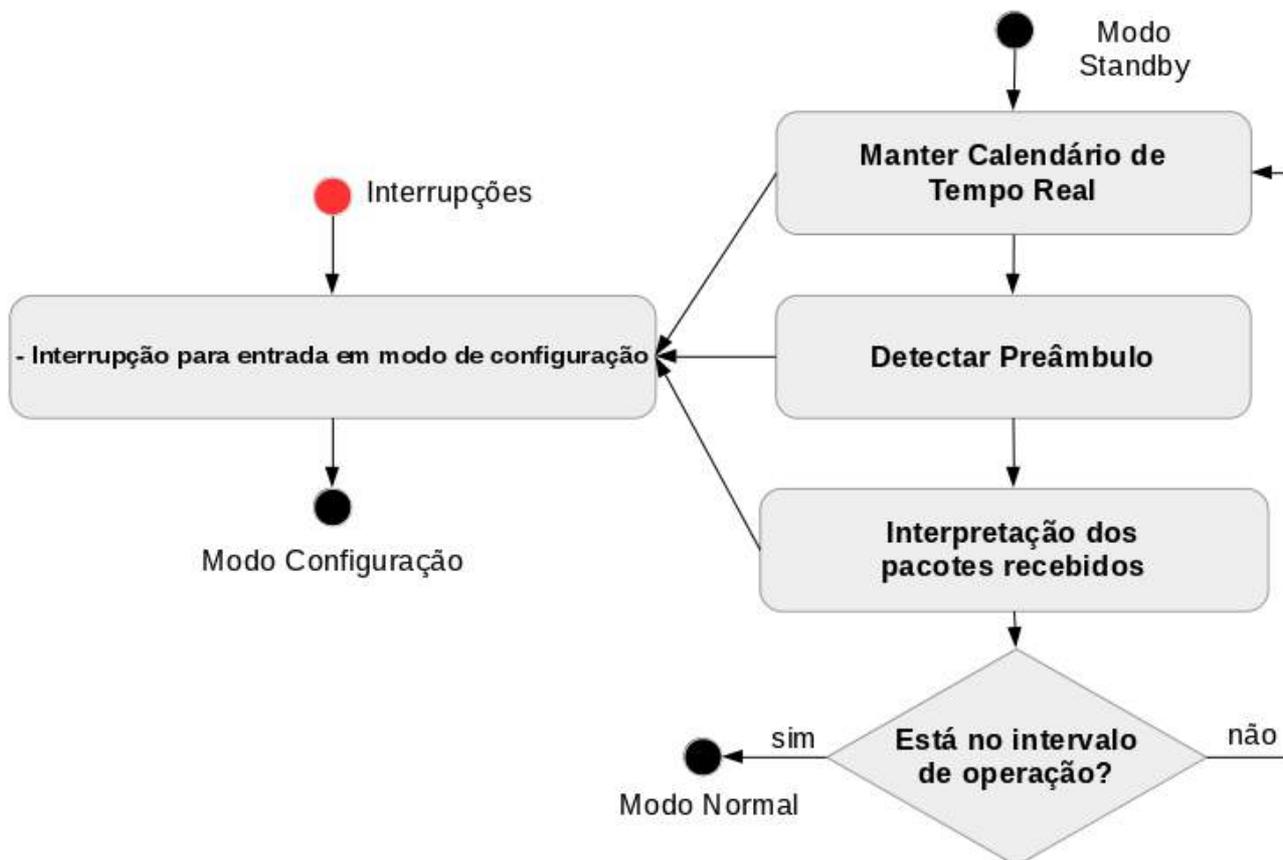


Figura 24: Diagrama de atividades do modo de operação *standby*

#### 4.4 – Software *Desktop*

Desempenhando papel importante no trabalho proposto, o software *desktop* proporciona a interface entre usuário e software embarcado. Através do software *desktop* o usuário poderá configurar os parâmetros requeridos para o funcionamento do seu sistema específico.

Para o desenvolvimento do software *desktop* utilizou-se a linguagem de

programação Java associado ao ambiente de desenvolvimento Eclipse.

Nesta seção apresenta-se o diagrama de casos de uso do software desenvolvido e sua interface mediante as operações por ele realizadas. Configurar o motor, RTC, sensores, gerar gráficos, são algumas das principais funções fornecidas pelo software *desktop*. No diagrama de casos de uso da Figura 25, construído para a modelagem do software, pode-se visualizar todas as ações que o usuário pode dispor através do software.

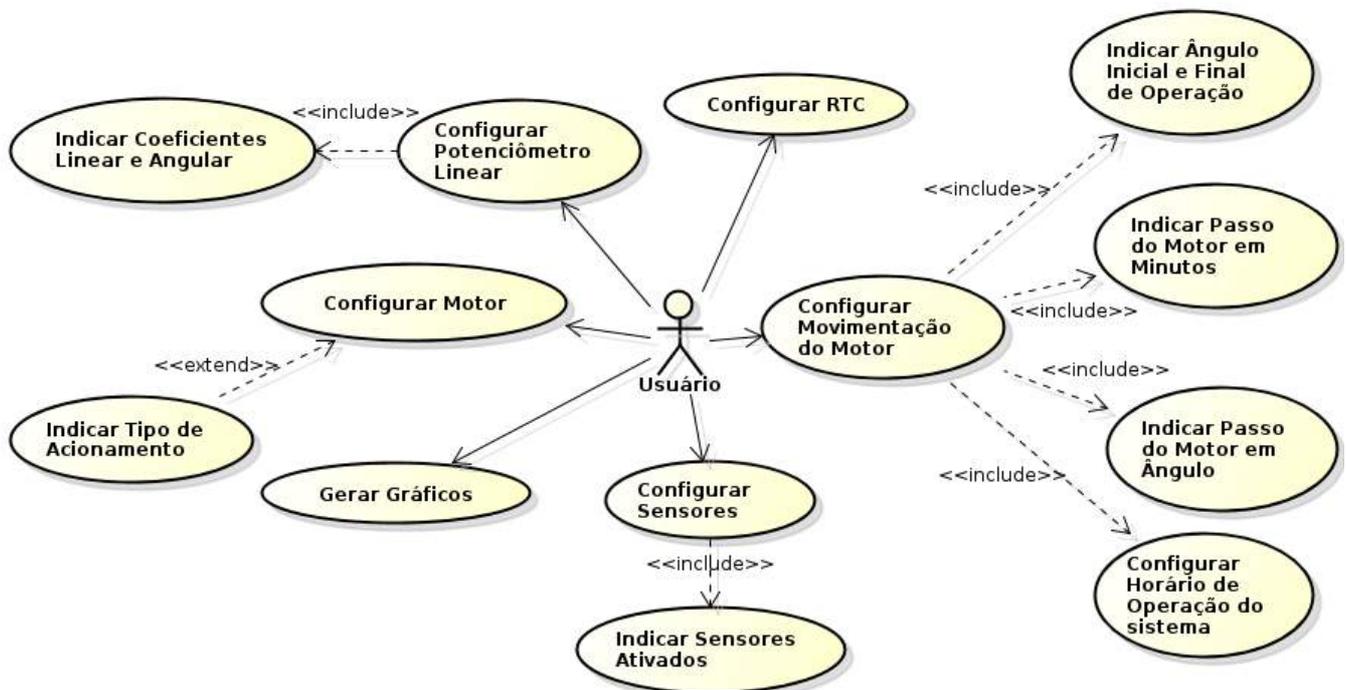


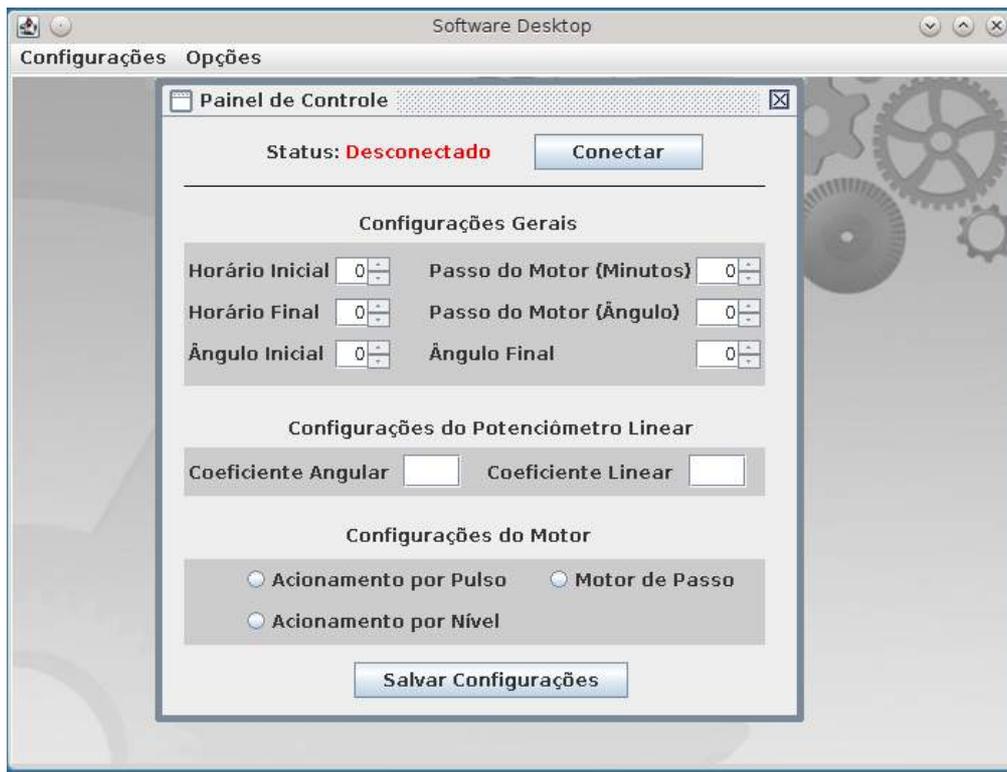
Figura 25: Diagrama de Casos de Uso

A tela inicial do software pode ser vista na Figura 26 abaixo. Vale ressaltar que a característica genérica do software possibilita a personalização do título inicial de acordo com a aplicação.



*Figura 26: Tela Inicial Software Desktop*

A partir da barra de menus o usuário pode acessar os recursos fornecidos pelo software. Na guia configurações o usuário poderá configurar o horário do RTC ou seguir para a tela do painel de controle. Através do painel de controle, que pode ser visto na Figura 27, o usuário pode indicar todas as informações necessárias para o início do funcionamento de todo o sistema.



*Figura 27: Painel de Controle*

O campo salvar informações, indicado na Figura 27, envia todas as informações setadas no painel de controle para o software embarcado, desde que o microcontrolador esteja conectado.

Na guia opções, além do usuário gerar gráficos e personalizar o título inicial do software, pode-se indicar quais dos sensores estarão ativados durante a operação do sistema embarcado. Tal informação pode ser vista na Figura abaixo.



*Figura 28: Sensores Ativados*

Assim como o título inicial, pode-se personalizar os nomes dos sensores. Desta maneira ao invés da utilização da nomenclatura sensor 0-1-2, utilizaria, por exemplo, sensor temperatura-luminosidade.

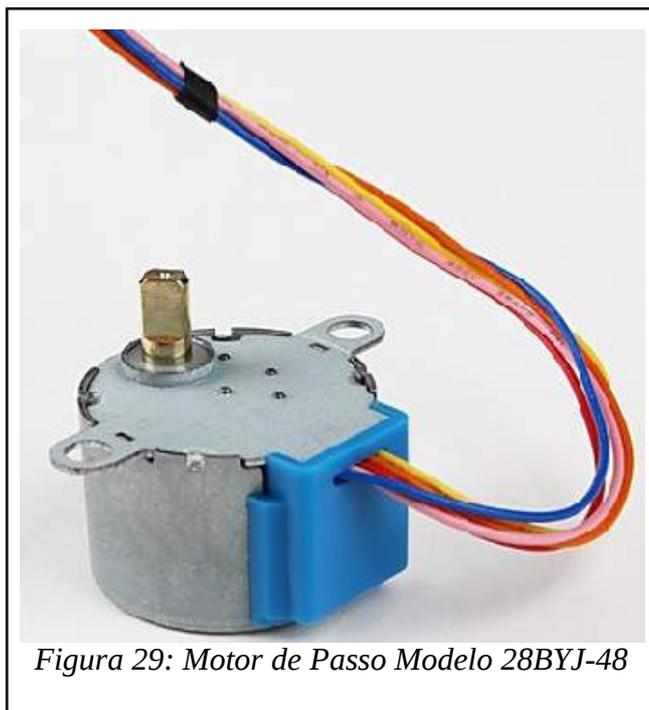
## CAPÍTULO 5 – RESULTADOS E DISCUSSÕES

Com o objetivo de validar experimentalmente o modelo desenvolvido nos capítulos anteriores, utilizou-se como ferramenta um painel fotovoltaico com concentrador solar e alguns testes com integração em motores de passo.

No sistema fotovoltaico, a plataforma foi empregada de modo a controlar o posicionamento do motor elétrico e conseqüentemente realizar o acompanhamento solar. Com o objetivo de validar a plataforma em outras configurações, realizou-se a integração junto a um kit composto por um motor de passo de baixa tensão. Assim, através do sistema fotovoltaico e dos kits, foi possível observar o funcionamento experimental da plataforma proposta de acordo com o projetado.

### 5.1 – Testes e Validação Utilizando Motores de Passo

Com o objetivo de realizar testes na plataforma enquanto a estrutura do painel fotovoltaico não encontrava-se disponível, criou-se um kit para o controle de posicionamento de motores, neste caso, um motor de passo de baixa tensão (5 – 12 Volts) modelo 28BYJ-48. O motor pode ser visto na Figura 29 abaixo.



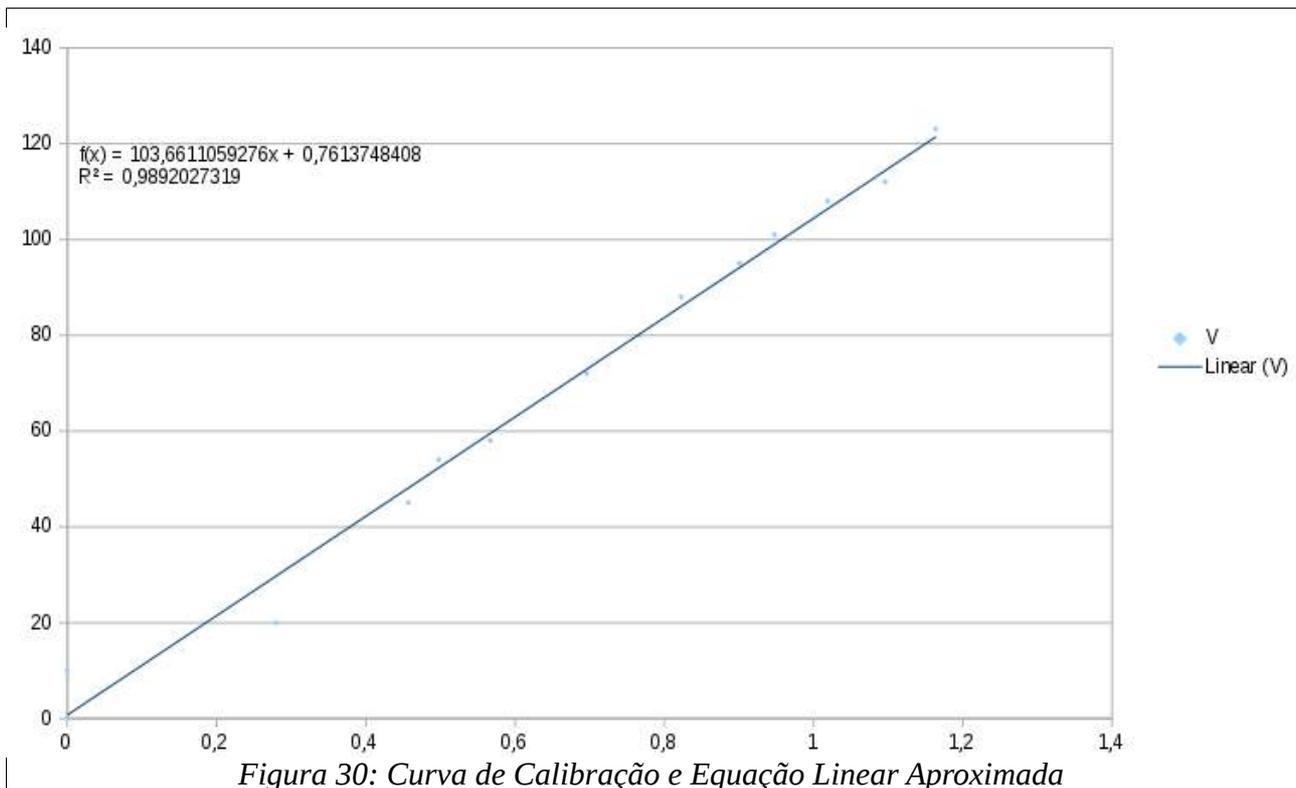
*Figura 29: Motor de Passo Modelo 28BYJ-48*

Para iniciar a integração, coletou-se inicialmente os dados referentes ao potenciômetro linear utilizado para o sistema de realimentação. O potenciômetro escolhido possuía resistência variável entre 0 e 10 K ohms com alimentação de 3 Volts. Nessas condições obteve-se os dados inseridos na tabela abaixo.

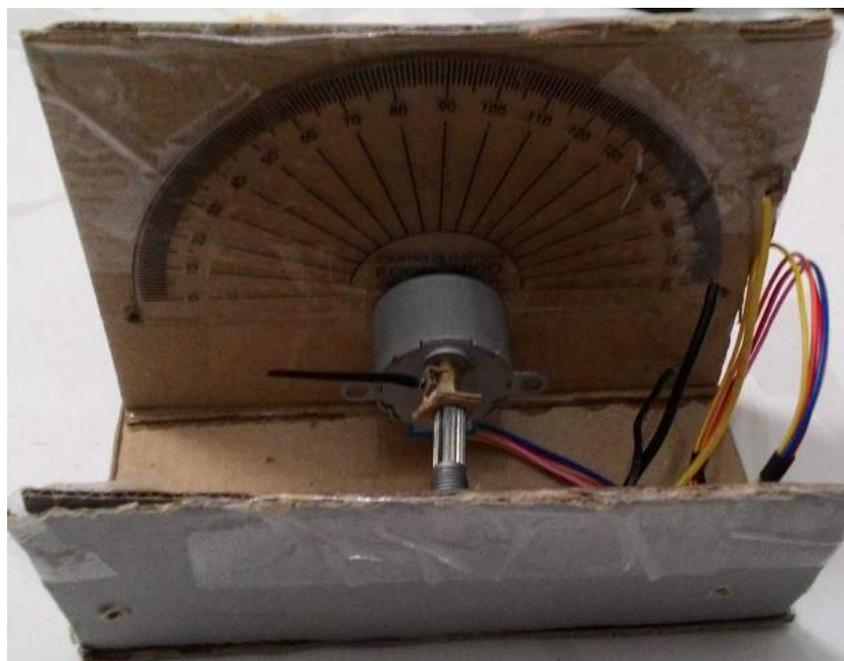
Ângulo	Tensão
20	0.281
45	0.430
54	0.498
58	0.567
72	0.696
80	0.767
88	0.823
95	0.901
101	0.948
108	1.019
112	1.096
123	1.164

*Tabela 4: Tabela Ângulo-Tensão do Potenciômetro*

Com as informações da tabela acima foi possível gerar a curva de calibração do potenciômetro quando alimentado em 3 Volts. Uma equação linear aproximada para a conversão de tensão em ângulo também foi gerada a partir dos dados, como pode ser vista na Figura abaixo.



A integração do sistema foi realizada como mostrado na Figura 15 do Capítulo anterior. O kit pode ser visto na Figura abaixo.



*Figura 31: Kit de Testes da Plataforma*

Em testes realizados durante alguns dias de operação, utilizando os mesmos

parâmetros de um dia de operação do painel fotovoltaico, observou-se que a plataforma realizou o posicionamento conforme o projetado. Verificou-se também que os modos de operação do sistema foram acionados nos momentos corretos. Os parâmetros utilizados nos testes pode ser visto na Figura a seguir.

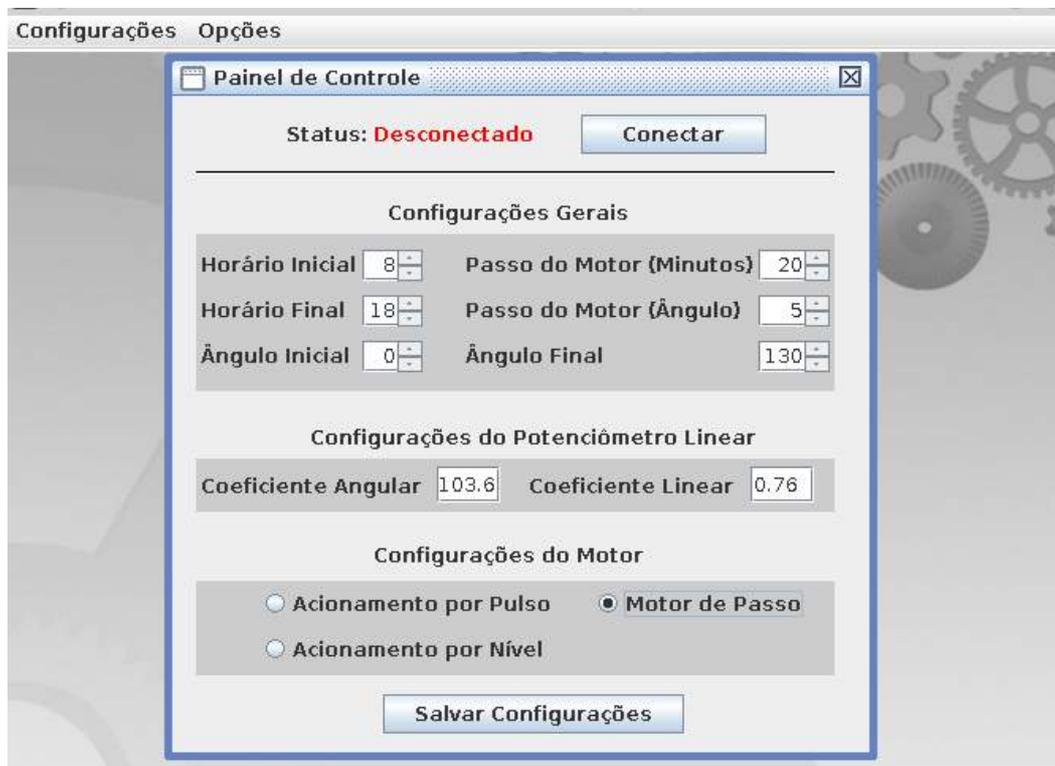
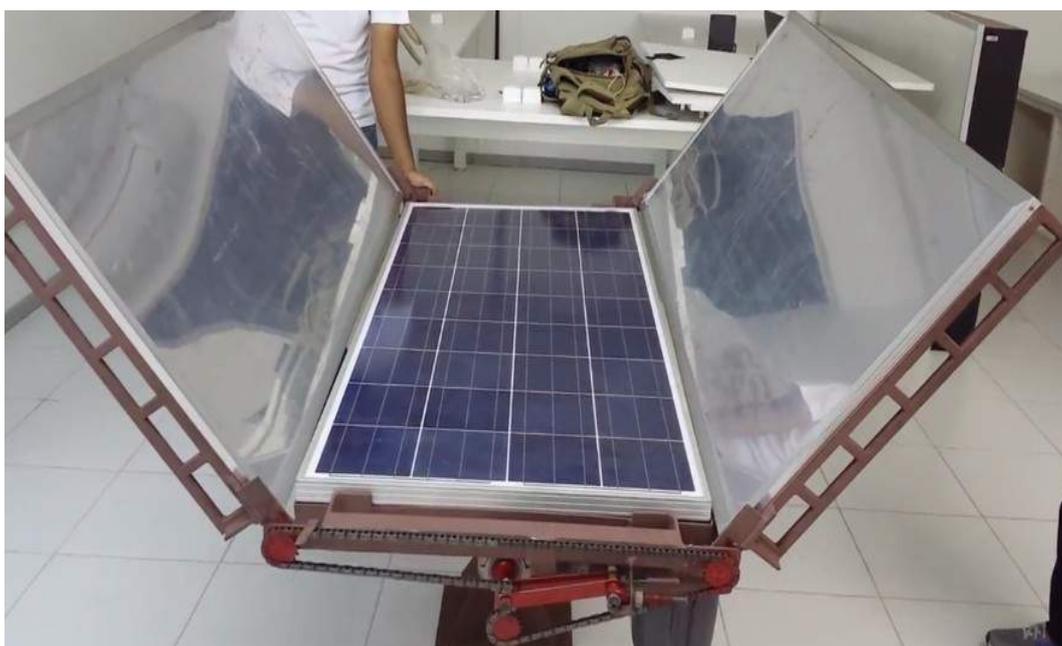


Figura 32: Configuração Software Desktop

## 5.2 – Validação com o Painel Fotovoltaico

O painel fotovoltaico com uso de concentradores, foi resultado de estudos realizados pelos componentes do grupo de pesquisa que este trabalho está inserido. Como consequência direta das pesquisas realizadas pelo grupo e de diversas reuniões de discussões de ideias, convergiu-se para um painel fotovoltaico de concentração e rastreamento solar, onde, apenas os concentradores se movem. O painel pode ser visto na Figura 33 abaixo.



*Figura 33: Painel Fotovoltaico*

Para a validação da plataforma, foram disponibilizados além do painel fotovoltaico, o motor elétrico modelo dz nano turbo Rossi com o intuito de o equipamento realizar a movimentação dos concentradores. Este motor elétrico é amplamente utilizado no âmbito residencial para controle de portões, algumas de suas características são a alimentação de 110 V, potência de 0,19 Kw com 1260 rpm e *driver* para acionamento do motor.

Como o *driver* acoplado ao dz nano turbo é o responsável por intermediar o acionamento do motor, fez-se necessário a realização de testes em laboratório com o objetivo de mapear o modo de acionamento do motor. Na Figura 34 abaixo é possível visualizar o *driver*.



Figura 34: Driver acionamento dz nano turbo Rossi

A partir dos testes realizados em laboratório, constatou-se que a ativação do motor por intermédio do *driver* era realizada no modelo de ativação por pulso. Neste modelo, um pulso é responsável por iniciar e finalizar a ativação do motor. Para a geração dos pulsos verificou-se a necessidade de utilizar uma saída tristate do microcontrolador, pois, o pulso só será detectado quando houver a transição entre 0 Volts e Alta impedância. O esquemático de ativação dos pulsos pode ser visto na Figura a seguir.

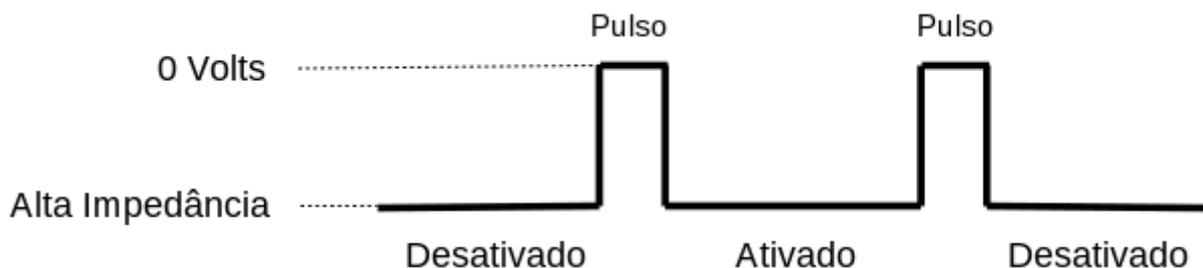


Figura 35: Tipo de Acionamento dz nano turbo

Com o conhecimento sobre o tipo de acionamento do motor, pode-se então realizar a integração entre a plataforma e o painel fotovoltaico. A integração de todo o sistema,

pode ser vista na Figura 36 abaixo.

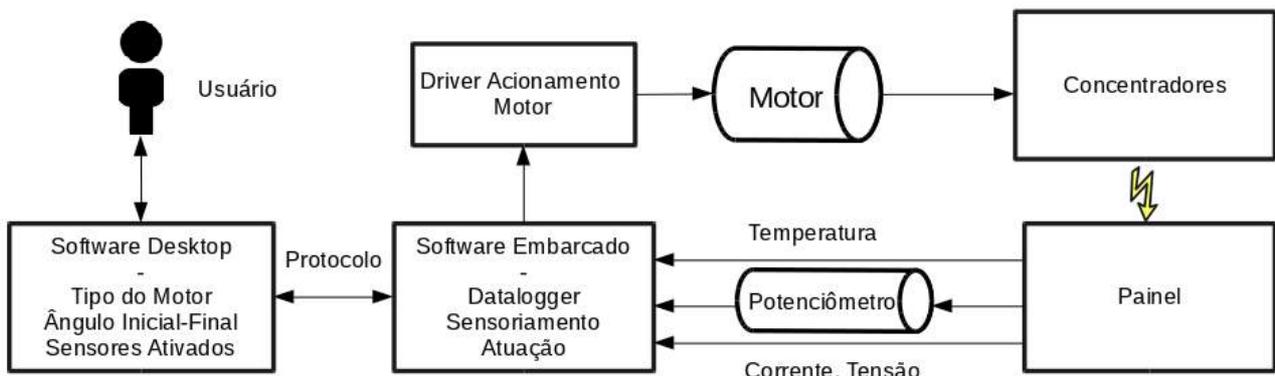


Figura 36: Integração do sistema com realimentação

O intuito inicial para a integração do sistema era a utilização da plataforma com a opção de realimentação, pois, haveria uma maior precisão no posicionamento. Além disso, pretendia-se observar o funcionamento do painel integrado a plataforma proposta durante um dia de operação. No entanto, uma divergência entre a estrutura de rastreamento projetada e a fabricada inviabilizou o acoplamento de um potenciômetro para o fornecimento do real posicionamento do painel. Outro fator que impactou negativamente na realização dos experimentos foi o atraso na entrega da estrutura, que foi entregue com aproximadamente 4 meses de atraso, reduzindo de forma considerável o tempo disponível para a integração da plataforma proposta com o painel fotovoltaico.

Sem a utilização de realimentação, a integração final pode ser vista na Figura abaixo.

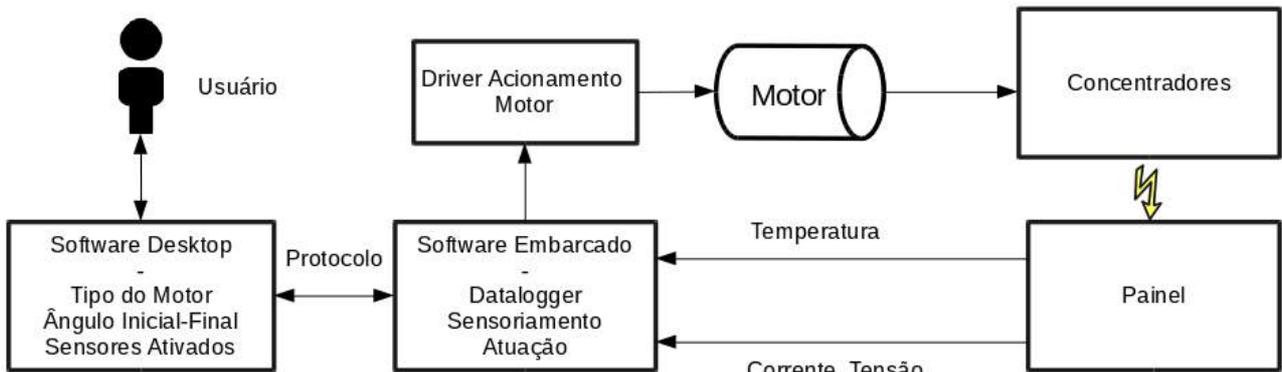


Figura 37: Integração final do sistema

Através do software *desktop*, configurou-se os parâmetros necessários para o funcionamento da plataforma. Como o tempo de operação foi reduzido, utilizou-se alguns parâmetros com valores também reduzidos quando comparados aos valores de um dia de operação, como, por exemplo, passo do motor em minutos e passo do motor em ângulo. A configuração realizada pode ser vista de acordo com a Figura a seguir.

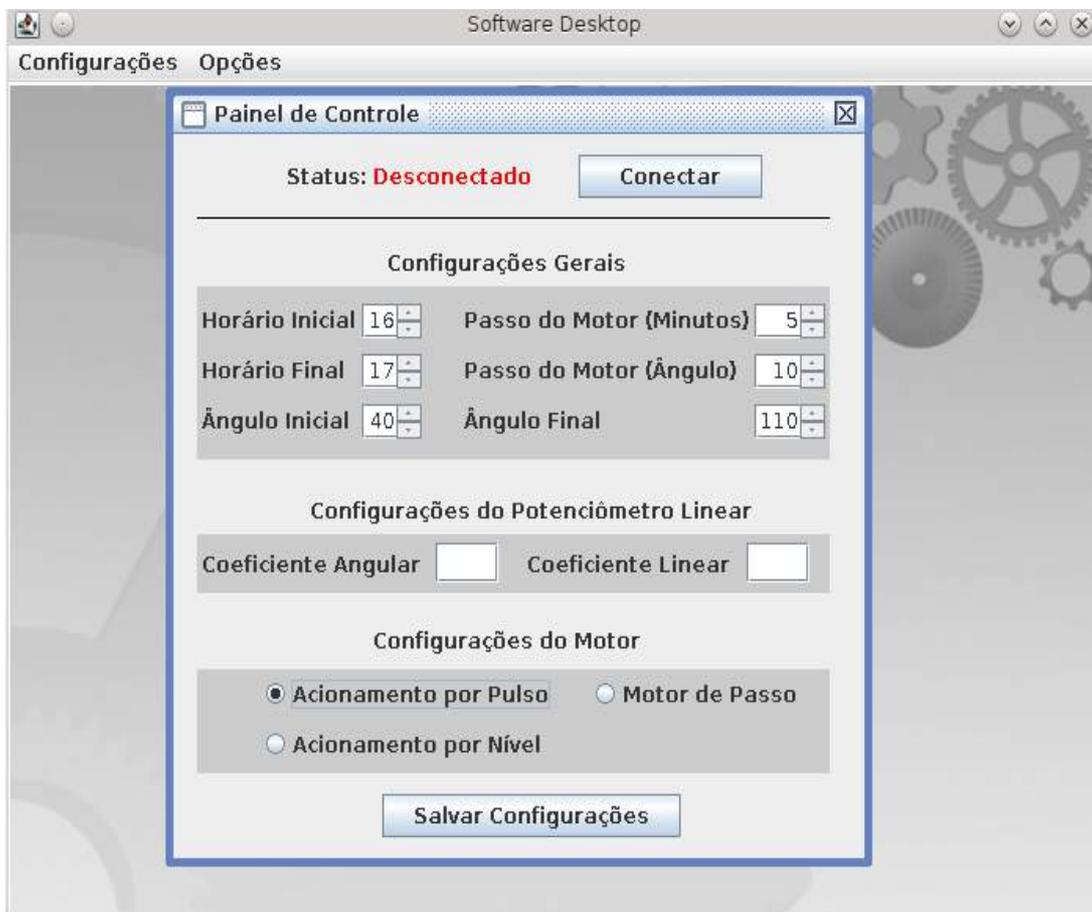


Figura 38: Configurações Software Desktop

Após indicar o horário do RTC, configurou-se a plataforma para realizar o posicionamento durante 1 hora com mudança de 5 graus a cada 1 minuto. Na Figura 39 abaixo pode ser visto o momento da configuração do software *desktop*.



Figura 39: Configuração do software *desktop*

Com a confirmação das configurações feitas no software *desktop*, verificou-se que o protocolo de comunicação e o software *desktop* assimilaram os comandos de acordo com o projetado. Constatou-se também que o posicionamento dos motores foi realizado de acordo com os parâmetros informados, ou seja, a atualização do posicionamento sendo realizada de acordo com o requerido. Além disso, mediante a troca de mensagens entre software embarcado e *desktop*, foi possível observar que os modos de operação foram acionados conforme o previsto.

Embora a integração da plataforma proposta não tenha sido a ideal, pois, não houve o sistema de realimentação e a sua operação foi observada em tempo reduzido, assegura-se a validação da plataforma, tendo em vista o correto funcionamento do posicionamento do motor e conseqüentemente do concentrador solar mediante os

parâmetros fornecidos pelo software desktop-embarcado.

## CONCLUSÃO E PERSPECTIVAS FUTURAS

No contexto inicial houve a necessidade do estudo de soluções já utilizadas para rastreamento solar e posicionamento de motores elétricos em sistemas de forma geral. Com isso, evidenciou-se a relevância na área de rastreamento solar em painéis fotovoltaicos. Propôs-se então uma plataforma embarcada para posicionamento de motores elétricos com validação em um sistema de rastreamento solar.

Adquiriu-se embasamento teórico e prático sobre sistemas embarcados e microcontroladores de arquitetura ARM. Através dos documentos de especificação do projeto, protocolo e requisitos organizou-se a modelagem da plataforma proposta. Aliando os conceitos e informações supramencionadas, foi desenvolvida uma plataforma embarcada para posicionamento de motores, composta por software *desktop*, protocolo de comunicação e software embarcado.

Muito embora o tempo para a coleta dos resultados não tenha sido o ideal, visto, o atraso na disponibilidade da estrutura do painel fotovoltaico, a plataforma proposta apresentou resultados conforme o projetado ao longo do projeto. Verificou-se também a correta comunicação entre os componentes da plataforma, assim como características arquiteturais do software embarcado como os modos de operação e controle em malha fechada.

Como a plataforma proposta realizou suas funções de forma adequada, a sua utilização em sistemas controlados por motores elétricos oferece uma forma mais simples e intuitiva para o usuário configurar o sistema de acordo com a necessidade, sem utilização de programação ou configurações de microcontroladores.

Em função da metodologia utilizada e do êxito na criação da plataforma, pretende-se continuar no desenvolvimento da ferramenta. No software *desktop*-embarcado são previstas algumas melhorias, como, por exemplo, a criação da curva de calibração pelo próprio software. Outra melhoria para implementações futuras e já prevista no protocolo de comunicação, é a possibilidade de realizar várias configurações simultâneas para o motor de acordo com o horário especificado.

Além disso, na plataforma de modo geral, pretende-se realizar testes e validação em outros tipos de motores e testes de usabilidade com o usuário.

## REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, Marcelo. PINTO, Aimé. RODRIGUEZ, Mauricio. ZILLES, Roberto. SILVA, José. **Desenvolvimento de Protótipo de Seguidor Solar de Eixo Horizontal para Sistemas Fotovoltaicos Conectados à Rede**. 2013, Rio de Janeiro. VII Congresso de Inovação Tecnológica em Energia Elétrica.

CARDOSO, Rômulo. CARVALHO, Edson. **Controle do Ponto de Máxima Potência em Painéis Fotovoltaicos com Microcontrolador PIC**. 2012, Rio de Janeiro. Centro Universitário de Volta Redonda.

CARVALHO, Edson. **Uma Nova Abordagem De Rastreamento Do Ponto De Máxima Potência Em Painéis Fotovoltaicos**. 2012, São Paulo. Universidade de Taubaté.

CÚNICO, Maurício. **Posicionamento Automático de Antenas Parabólicas**. 2006, Paraná. Centro Universitário Positivo.

DUANY, Ernesto. MUNOZ, Javier. MORERA, Mario. **Aplicación de RT-Linux en el control de motores de pasos**. 2007, Instituto Superior Politécnico Jose Antonio Echeverría.

ECLIPSE. **Eclipse Foundation**. Disponível em: <<https://eclipse.org/projects/>>. Acesso em: 22 de fevereiro de 2015.

EPE. **Empresa de Pesquisa Energética**. Disponível em: <<http://www.epe.gov.br/Paginas/default.aspx>>. Acesso em: 10 de fevereiro de 2015.

FERREIRA, Alceu. **Desenvolvimento De Um Sistema De Posicionamento Automático Para Painéis Fotovoltaicos**. 2008, São Paulo. Universidade Estadual Paulista.

MAGALHÃES, Renato. **Modelagem, Simulação E Validação Experimental De Um Sistema De Rastreo De Potência Máxima Para Geradores Solares De Satélites Artificiais**. 2006, São Paulo. Instituto Nacional de Pesquisas Espaciais.

MME. **Ministério do Meio Ambiente**. Disponível em: <<http://www.mme.gov.br/programas/proinfa>>. Acesso em: 10 de fevereiro de 2015.

MOURA, Ricardo. **O Uso De Microcontroladores No Acionamento E Controle De Motores Brushless dc**. 2010, Escola de Engenharia de São Carlos.

NASCIMENTO, Cássio. **Princípio De Funcionamento Da Célula Fotovoltaica**. 2004, Minas Gerais. Universidade Federal de Lavras.

OPENOCD. **Free and Open On-Chip Debugging, In-System Programming and Boundary-Scan Testing**. Disponível em: <<http://openocd.org/>>. Acesso em: 22 de fevereiro de 2015.

RUZU, C. SZOKE, E. **Embedded System Design for Field-Oriented Stepper Motor Control**. 2006, Technical University of Cluj.

SALERNO, Carlos. CAMACHO, Roberto. OLIVEIRA, Antonio. **Controle de Velocidade do Motor de Indução Trifásico utilizando Microcontroladores para Aplicações em Baixa Potência**. 2000, Universidade Federal de Uberlândia.

SILVA, Roberto. PEREIRA, Maicon. SALVADOR, Caroline. CANCIAN, Rafael. TORRES, Roberto. ZEFERINO, Cesar. AstroFácil: **Sistema Computacional Embarcado para Automação de Telescópios de Pequeno Porte**. 2006, Santa Catarina. Universidade do Vale do Itajaí.

SOUSA, Gilberto. BOSE, Bimal. CLELAND, John. **Fuzzy Logic Based On-Line Efficiency Optimization Control of an Indirect Vector-Controlled Induction Motor Drive**. 1995, Universidade Federal do Espírito Santo.

SOUZA, José. **Projeto, Construção E Levantamento De Desempenho De Um Concentrador Solar Cilindro Parabólico Com Mecanismo Automático De Rastreamento Solar**. 2008, Natal. Universidade Federal do Rio Grande do Norte.

STINE, W. GAYER, M. **Power Of Sun**. Disponível em: <<http://www.powerfromthesun.net/>>. Acesso em: 14 de abril de 2015.

TEODORO, Silva. **Desenvolvimento de um sistema mecatrônico para posicionamento de um painel fotovoltaico e comparação com painel fixo**. 2010, Universidade Federal da Bahia.