

Sistema de Posicionamento para Placas Fotovoltaicas

Aplicação de microcontroladores para o melhoramento da eficácia de painéis solares

Jéssica de Souza Alves

Graduanda em Engenharia Eletrônica
Universidade de Brasília - UnB
Brasília, Brasil
14/0042784
jessicaturunenn@gmail.com

Lorena Albernaz

Graduanda em Engenharia Eletrônica
Universidade de Brasília - UnB
Brasília, Brasil
14/0025715
lorena.albernazz@gmail.com

Resumo— O projeto visa criar um sistema de rotação de placas fotovoltaicas de acordo com o posicionamento do sol, também conhecido como solar tracker, sendo este controlado por um MSP 430.

Palavras-chave: MSP, Seguidor Solar, Fotovoltaico.

1. Introdução

A energia solar é uma fonte limpa e sustentável de energia, e nos últimos tempos tem sofrido impacto devido ao seu crescimento. Dessa forma, métodos para o melhoramento de captação melhora consequentemente o aproveitamento e custos relacionados à energia desse tipo, sendo assim cada vez mais eficaz.

A energia solar fotovoltaica é a energia obtida através da conversão direta da luz solar em eletricidade, sendo a célula fotovoltaica, um dispositivo fabricado com material semicondutor, sendo essa célula fotovoltaica a unidade fundamental desse processo de conversão. A forma em que a célula fotovoltaica converte a luz é chamado de efeito fotovoltaico [1].

Uma célula solar ou célula fotovoltaica é um dispositivo elétrico de estado sólido capaz de converter a luz proveniente do sol (energia solar) diretamente em energia elétrica por intermédio do efeito fotovoltaico. [2]

Atualmente, as células fotovoltaicas apresentam eficiência de conversão da ordem de 16%. Até existem células fotovoltaicas com

eficiências de até 28%, fabricadas de arseneto de gálio, mas o seu alto custo limita a produção dessas células solares para o uso da indústria espacial. [3]

Uma boa maneira melhorar o rendimento de placas de captação de energia fotovoltaica, é criando dispositivos, que consideram o movimento do sol como um fator real, e fazem com que as placas solares sigam a direção em que a incidência solar se torna maior.

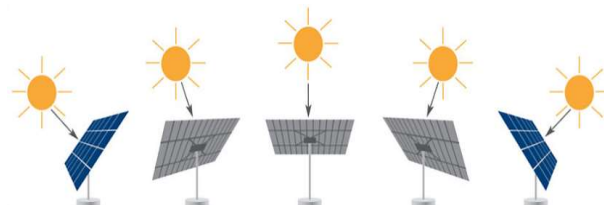


Figura 1 - Funcionamento simplificado de um seguidor solar

Seguidores Solares atuam na movimentação de placas fotovoltaicas adequando suas posições para o sentido de maior incidência solar de modo a otimizar a produção de energia da placa.

Sistemas com seguidores solares aumentam em torno de 30% a produção de energia quando comparados à sistemas fixos. Isso ocorre devido ao aumento da exposição direta aos raios solares. [4]

Como a implementação de um sistema fotovoltaico ainda é de alto custo, a utilização de um sistema de posicionamento de placas é uma boa alternativa para redução de custos de médio prazo.

A implementação do sistema de posicionamento é simples, barata e eficaz. Além disso, pode otimizar a produção de energia em até 30% , sendo assim, uma boa alternativa para sistemas solares instalados sobre lajes residenciais onde a rotação das placas é viável

O protótipo visa ter um custo mais atrativo do que os disponíveis no mercado, beneficiando a sociedade e aumentando o acesso à energia limpa.

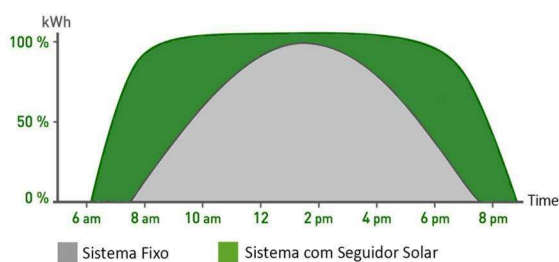


Figura 2 - Gráfico comparativo entre os sistemas fotovoltaicos fixos e com seguidor solar

2. Desenvolvimento

2.1 Descrição de Hardware

O Hardware do projeto se dará de forma geral por uma placa microcontroladora que irá controlar dois motores (cada um em um eixo), a partir de dados de luminosidade coletado por sensores.

Serão utilizados quatro sensores LDR's (Light Dependent Resistor). Os sensores serão utilizados para o posicionamento das placas, de forma que ao fazerem as coletas de luminosidade do ambiente, o software irá comparar os dados dos sensores, que irão fazer as coletas de intensidade luminosa do ambiente, e com os dados obtidos, dois motores recebem os comandos para movimentação.

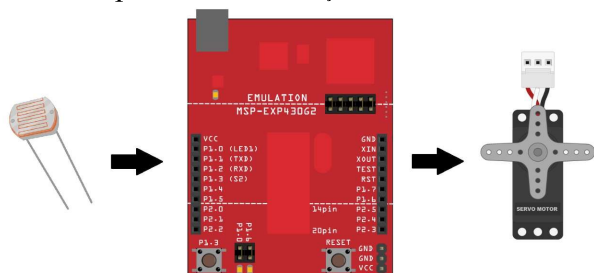


Figura 3 - Esquemático de Hardware do seguidor solar (da esquerda para a direita: sensores LDR, MPS430, Motores Servo).

2.2. Estrutura

A estrutura foi pensada para permitir a movimentação do sistema de posicionamento em dois eixos, conforme a Figura 4, sendo eficiente para instalação em qualquer lugar, independente do posicionamento geográfico ou da incidência solar do local.

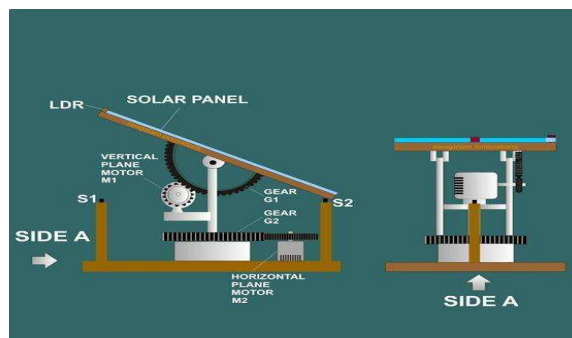


Figura 34- Possível mecanismo para um sistema de movimentação de um painel solar com dois eixos [5]

Para implementar a estrutura planejada, são necessários suportes para dois motores independentes, cada um para permitir a movimentação em um eixo (vertical e horizontal). Além dos motores, a estrutura contém um suporte para o painel fotovoltaico, espaço para posicionamento dos sensores, engrenagens que permitem movimentação e uma base de apoio.

A estrutura do projeto foi impressa por uma impressora 3D e as peças foram planejadas separadamente, como ilustradas a seguir, e depois encaixadas para devida implementação.

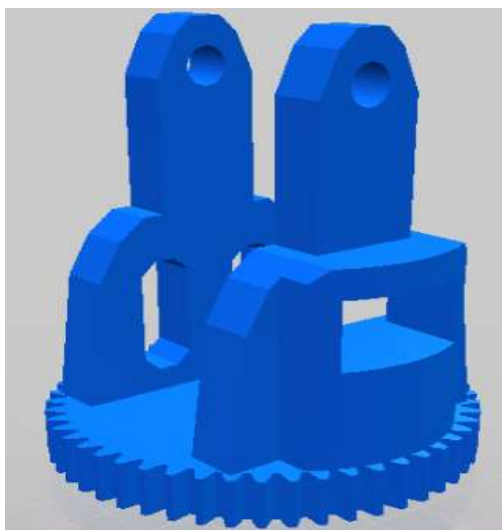


Figura 5 - CAD do mecanismo de rotação horizontal

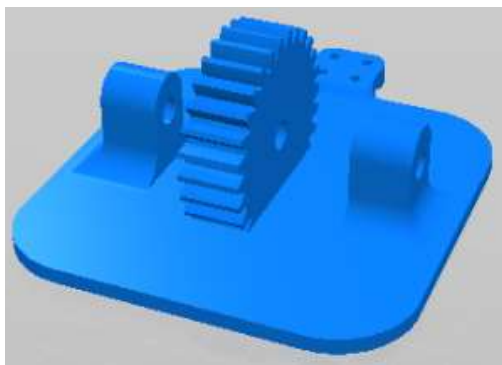


Figura 6 - CAD do suporte do painel solar e LDR

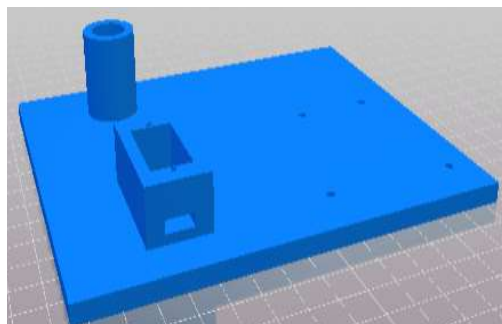


Figura 7 - CAD da base da estrutura

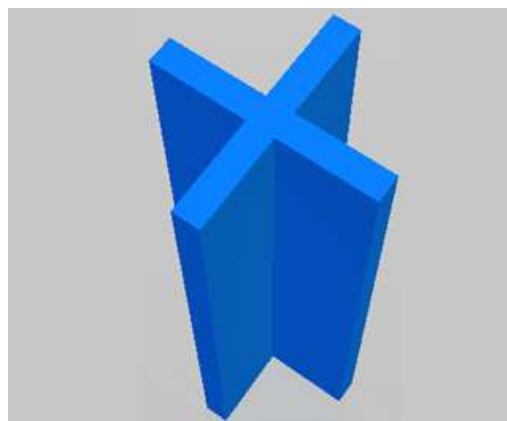


Figura 8 - CAD do divisor de LDRs

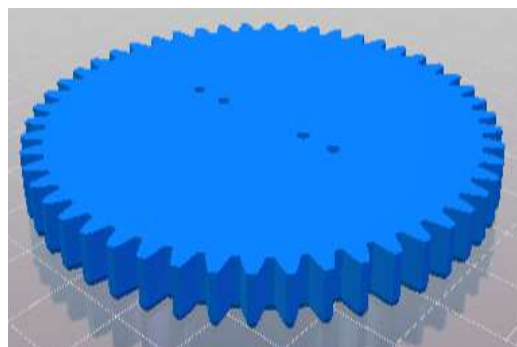


Figura 9 - CAD da engrenagem de movimentação horizontal

2.3 Descrição de Software

Cada LDR no circuito funciona como um divisor de tensão ao ser montado com um outro resistor. A saída deste divisor de tensão é atribuída a um pino do microcontrolador e então variáveis serão atribuídas para guardar o valor de cada um dos quatro LDR's.

Para acionar o microcontrolador, são utilizados dois conjuntos com dois LDR's cada um. Um conjunto controla o movimento azimuthal e outro conjunto controla o ângulo de declinação.

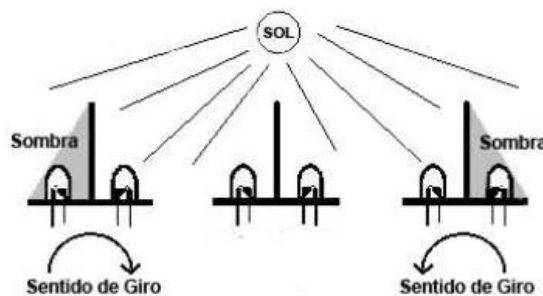


Figura 10 - Perfil "T" separando os LDR's de um conjunto

Os LDR's de cada conjunto são separados um do outro por meio de um perfil "T" conforme a Figura 10. O microcontrolador é acionado quando existe uma diferença entre as impedâncias dos LDR's. O microcontrolador envia um pulso para movimentação (PWM) ao motor quando nota essa disparidade e reposiciona o sistema até que a discrepância entre as impedâncias seja mínima.

Um dos conjuntos de LDR's compara a intensidade luminosa entre o lado direito e o lado esquerdo enquanto o outro conjunto compara os valores de um referencial em cima e outro abaixo conforme os referenciais da Figura 11. Após a comparação, é feita uma média com cada dado, para o microcontrolador acionar os motores.

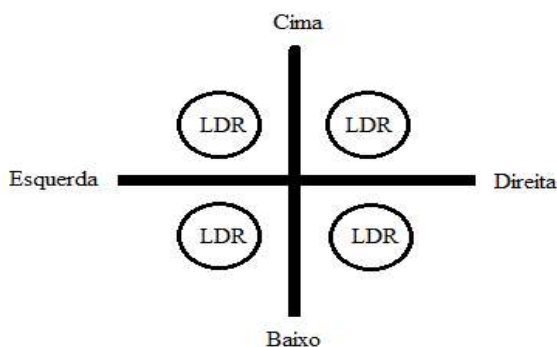


Figura 11 -- Referencial de orientação utilizado

Após cada análise nos valores de incidência de luminosidade de cada conjunto os motores são movimentados até que a discrepância entre os valores analisados seja mínima.

2.4. Lista de Materiais

- MSP430
- 4 resistores de 10 k Ohms
- 2 servo motores SG90
- estrutura impressa
- 4 LDR's
- cabos e fios para conexões

2.5 Descrição do código utilizado no MSP

O sistema foi montado e testado conforme o esquemático da Figura 12 utilizando 4 LDR's, dois servo motores SG90, 4 resistores de 10K

ohms e um MSP430. mplementado no Studio Code Composer.

A lógica de funcionamento dos LDR's para o controle dos motores conforme explicada na descrição de software foi validada no teste do protótipo.

Após o teste e validação do protótipo, as alterações foram feitas para que houvesse o devido funcionamento no MSP430, sendo o código compilado no *Code Composer Studio*.

Para analisar os valores vindos dos LDR's, foi utilizado o ADC10AE0 (conversor analógico\ digital de até 10 bits) que recebem os dados dos quatro pinos de entrada (SENS_ESQUERDA, SENS_DIREITA, SENS_BAIXO, SENS_CIMA)

Os dados dos sensores, após conversão digital são atribuídas aos bit 0, bit, 1, bit 2 e bit 3.

No início da função main (onde são definidos os detalhes dos pwm dos servos) é possível reparar que o watchdog time foi desligado enquanto TA0CCTL1 e TA1CCTL1 setam e, ou, resetam o TACCR0 e TACCR1 respectivamente.

O ConfigureAdc() configura a maneira que a conversão analógico digital deve funcionar, a princípio, tem um comando de espera enquanto a conversão é feita, em seguida vem a função que incrementa os servos mecanismos possibilitando seu movimento onde os valores de TACCR0 e TACCR1 são incrementados.

Para movimentação dos motores, o sample é utilizado, pois já pega o valor resultante do conversor AD para definir os comandos . Por fim, o código conta com uma função de interrupção do movimento dos motores.

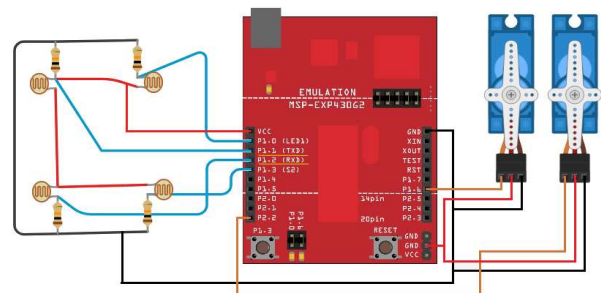


Figura 12 - Esquemático de montagem definitivo

3. Resultados

Após a montagem de todos os sistemas, incluindo toda a estrutura impressa, já com os motores encaixados e com o microcontrolador conectado (Figura 13), foram feitos testes para validar o funcionamento.

Os testes constataram que o sistema de posicionamento funciona adequadamente, porém devido a calibragem dos sensores, a melhor performance foi verificada apenas em ambientes com penumbra ou mais escuros, e nem sempre funcionando adequadamente à exposição direta a intensidade luminosa do sol.

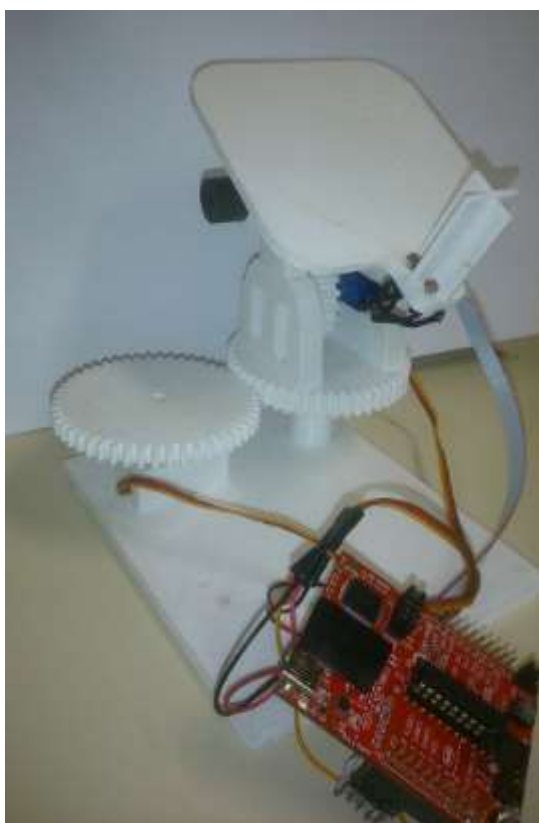


Figura 13 - Estrutura pronta

4. Conclusões

Com o protótipo em funcionamento não foi possível mensurar se o percentual de energia gerada aumenta pois não se teve acesso a uma placa fotovoltaica, porém, diversas bibliografias (como a citada na referência [4]) indicam que a melhora de fato ocorre.

O sistema de posicionamento para painéis fotovoltaicos funciona adequadamente, apesar da calibragem do sistema não ter sido feita para a luz solar.

Conclui-se então, que o sistema feito pode ser melhor descrito como uma possível base de testes para se poder averiguar numericamente qual a real melhora que o movimento do seguidor solar de fato exercita na eficiência de painéis fotovoltaicos.

5. Referências

- [1] Manual de engenharia para sistemas fotovoltaicos. Rio de Janeiro: CEPEL - CRESESB. Março de 2014.
- [2] Roper, L. David (24 de agosto de 2011). World Photovoltaic Energy.
- [3] Água quente solar - Caracterização do Solar Térmico em Portugal 2008. Agência para a Energia.
- [4] Disponível em <http://www.portalsolar.com.br/blog-solar/painel-solar/seguidor-solar---tracker-vantagens-e-desvantagens-parte-1.html>
- [5] Disponível em <http://www.brighthub.com/environment/renewable-energy/articles/76226.aspx>.
- [6] GOETZBERGER, A.; HEBLING, C.; SCHOCK, H.-W. Photovoltaic materials, history, status and outlook. Materials Science and Engineering R, v. 40, p. 1-46, 2003.
- [7] GREEN, M. A. Photovoltaics: technology overview. Energy Policy, v. 28, p. 989-998, 2000.
- [8] OLIVEIRA, Fernanda P. Monografia: Célula Fotovoltaica de Silício. São Paulo: USP, 2010

4. Anexos

Código do projeto (Feito no Code Composer Studio):

```
#include "msp430.h"
#define ADC_CH 4
unsigned int samples[ADC_CH];

#define SENS_ESQUERDA BIT0
#define SENS_DIREITA BIT1
#define SENS_CIMA BIT2
#define SENS_BAIXO BIT3

#define CLOCK 1000000
#define PWM_F 46 // frequência do pwm
#define SERVO 30 //quantidade máxima de passos em graus

unsigned int PWM = (CLOCK/ PWM_F);
unsigned int PWM_Duty = 0;
int i;
int count;
int count2;

void ConfigureAdc(void){

ADC10CTL1 = INCH_3 | ADC10DIV_0 | CONSEQ_3 | SHS_0;
ADC10CTL0 = SREF_0 | ADC10SHT_2 | MSC | ADC10ON | ADC10IE;

ADC10AE0 = SENS_ESQUERDA + SENS_DIREITA + SENS_CIMA +
SENS_BAIXO;
ADC10DTC1 = ADC_CH;
}

void main(void) {

WDTCTL = WDTPW | WDTHOLD;
TA1CCTL1 = OUTMOD_7;
TA0CCTL1 = OUTMOD_7;
TA1CTL = TASSEL_2 + MC_1;
TA0CTL = TASSEL_2 + MC_1;
TA1CCR0 = PWM-1; // periodo do pwm
TA0CCR0 = PWM-1;
TA1CCR1 = 0; duty cycle
TA0CCR1 = 0;

P1DIR = 0;
ISEL = 0;
IOUT = 0;

IREN |=
(SENS_ESQUERDA|SENS_DIREITA|SENS_BAIXO|SENS_CIMA);

P1DIR |= BIT6;
P2DIR |= BIT2;
P1SEL |= BIT6;
P2SEL |= BIT2;

ConfigureAdc();
__enable_interrupt();
while (1) {
    __delay_cycles(1000);
    ADC10CTL0 &= ~ENC;
    while (ADC10CTL1 & BUSY);
    ADC10SA = (unsigned int)samples;
    ADC10CTL0 |= ENC + ADC10SC;

    __bis_SR_register(CPUOFF + GIE);

    for (i = 0; i < SERVO; i++) {
```

```
        TA1CCR1 = count2;
        TA0CCR1 = count;
        __delay_cycles(20000);
        TA1CCR1 = count2;
        TA0CCR1 = count;
        __delay_cycles(20000);
    }
}

#pragma vector = ADC10_VECTOR
__interrupt void ADC10_ISR (void){

if (samples[0] < samples[1]){
    if(samples[1]> 550){
        count = 850;
    } else {
        count = 1350;
    } } else if ((samples[0] + samples[1])/2 < 500) {
        count = 1350;
    } else {
        if(samples[0]>550){
            count = 1850;
        } else {
            count = 1350;
        }
    }
}

if (samples[2] < samples[3]) {
    if(samples[3]> 550){
        count2 = 850;
    } else {
        count2 = 1150;
    } } else if ((samples[2] + samples[3])/2 < 500) {
        count2 = 1150;
    } else {
        if(samples[2]>550){
            count2 = 1850;
        } else {
            count2 = 1150;
        }
    }
}

__bic_SR_register_on_exit(CPUOFF);
}
```