

Rastreador de luz para painéis solares

Adolfo de Souza Serique, 140127640

Universidade de Brasília, Faculdade Gama
Microprocessadores e Microcontroladores, 201383
Brasília, Brasil
adolfoserique@gmail.com

Resumo — projeto que melhora a eficiência da captura de energia por placas solares por meio de um sistema que rastreia a luz do sol e rotaciona a placa de acordo com o movimento do sol.

I. INTRODUÇÃO

Devido a atual crise energética e ambiental em que o mundo se encontra, faz-se necessário o desenvolvimento de novas fontes que sejam sustentáveis e de baixo custo.

Uma promissora fonte de energia não convencional é a energia solar, que oferece uma grande perspectiva para conversão em energia elétrica, o qual pode garantir grande parte da demanda energética do planeta.

A conversão fotovoltaica é baseada no em converter a luz solar em eletricidade. Usando uma tecnologia de rastreamento a produção do painel solar pode aumentar entre 30%-60% diferentemente das instalações fixas que costumam a ser apenas por volta de 24%.

Utilizando um circuito eletrônico micro-controlado para rastrear a luz solar e maximizar a incidência de luz nos painéis, aumentando o débito dos módulos solares. Um sistema portanto que coleta a luz do Sol por mais tempo, já que ele se movimenta de acordo com a posição do astro.

O projeto deve rastrear a área de maior intensidade de luz por meio de sensores para controlar por meio de motores o painel solar a fim de maximizar a produção de energia.

O desenvolvimento de tecnologias de fontes de energia solar acessíveis e mais eficientes por meio do uso do projeto aqui realizado terá enormes benefícios a longo prazo. Aumentará a segurança energética dos países através da dependência de um recurso inesgotável e, principalmente, independente de importação, o que aumentará a sustentabilidade, reduzirá a poluição, reduzirá os custos de mitigação das mudanças climáticas e manterá os preços dos combustíveis fósseis mais baixos. Estas vantagens são globais. Sendo assim, entre os custos adicionais dos incentivos para a implantação precoce dessa tecnologia devem ser considerados investimentos em aprendizagem; que deve ser gasto com sabedoria e precisam ser amplamente compartilhado

Luana Carolina de Val Abreu, 140150234

Universidade de Brasília, Faculdade Gama
Microprocessadores e Microcontroladores, 201383
Brasília, Brasil
luanacarolinadva@gmail.com

II. DESENVOLVIMENTO

A. Hardware

O hardware foi projetado em pequenos blocos como representado na figura 1. Cada um desses blocos são detalhados nas subseções seguintes.

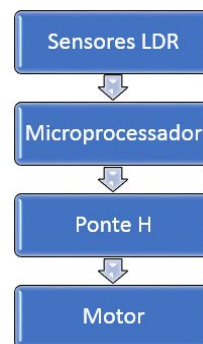


Figura 1: Diagrama de blocos do hardware

1) Sensor de luz

Para identificar os níveis de luz fornecidos pelo sol foi idealizado um sistema que utiliza sensores de luz (LDR) acoplado juntamente a um potenciômetro.



Figura 2: Sensor LDR

LDR (do inglês Light Dependent Resistor), em português Resistor Dependente de Luz é um componente eletrônico passivo do tipo resistor variável, mais especificamente, é um resistor cuja resistência varia conforme a intensidade da luz (iluminamento) que incide sobre ele. Tipicamente, à medida que a intensidade da luz aumenta, a sua resistência diminui. O sensor é construído a partir de material semicondutor com elevada resistência elétrica. Quando a luz que incide sobre o semicondutor tem uma frequência suficiente, os fótons que incidem sobre o semicondutor liberam elétrons para a banda condutora que irão melhorar a sua condutividade e assim diminuir a resistência. O potenciômetro teria a função de

calibrar o sensor afim de obter melhores resultados.

2) Motor DC

A escolha de um motor DC simples foi feita pelas seguintes vantagens:

- Operação em corrente contínua;
- Constante de reversão;
- Velocidade ajustável;
- Alto torque de partida.

Um motor de 5V foi conectado nos pinos 3 e 6 da H-Bridge para movê-lo de acordo com os sinais recebidos do microcontrolador MSP 430.

3) L293D H-Bridge

Devido aos requisitos de eficiência energética desse projeto foi implementada também uma ponte H que tem como principal função o controle de velocidade e sentido de motores DC. O circuito necessita de um caminho que carregue a corrente ao motor em uma direção, e outro caminho que leve a corrente no sentido oposto. Além disso, o circuito deve ser capaz de ligar e desligar a corrente que alimenta o motor. Uma ponte H possui quatro interruptores eletrônicos, que podem ser controlados de forma independente.



Figura 3: Ponte H

O CI L293D foi escolhido pelas seguintes vantagens:

- Tensão lógica de entrada separada;
- Tensão de alimentação entre 4.5V;
- Shutdown térmico que se desliga quando muito quente.

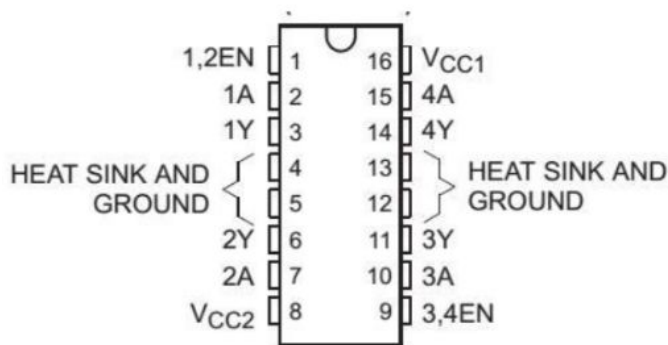


Figura 4: Diagrama do L293D

A figura 5 mostra o esquema para a montagem do circuito.

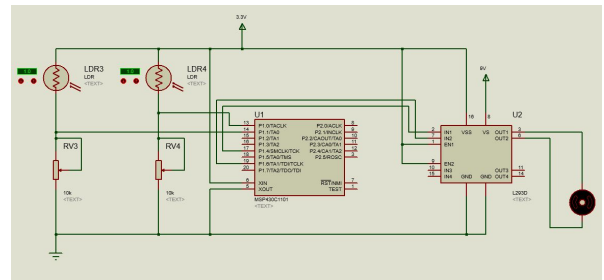


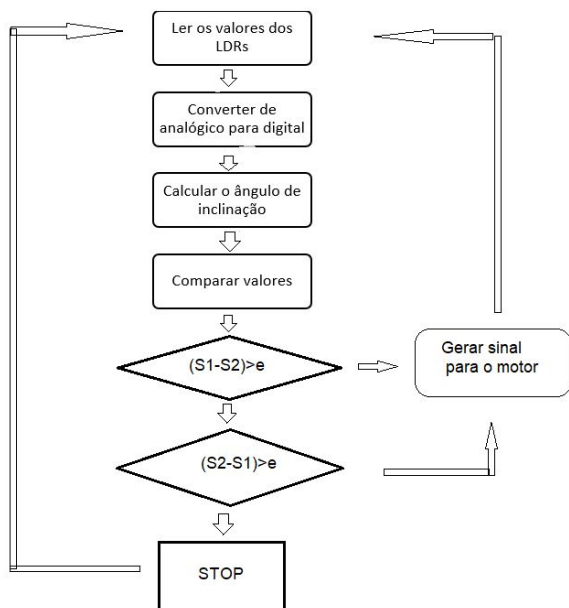
Figura 4: Esquemático do circuito

B. Software

Para o software, primeiramente foram definidas as entradas e saídas, que são respectivamente os sinais obtidos pelos LDRs e o sinal mandado para H-Bridge. Em seguida os sinais obtidos dos sensores passam por um sistema de conversão AD. Nos sensores de LDR quanto maior a luz incidida sobre ele maior sua resistência então foi implementado um sistema que compara a resistência do sensor esquerdo com o direito e acende o led (no MSP430) que representa esse lado.

O controle do motor pode ser descrito pelo algoritmo abaixo:

- Entrada das voltagens dos dois LDRs;
- Conversão desses valores analógicos para digitais;
- Comparação desses dois valores e obter a diferença entre eles.
- Fazer dessa diferença o ângulo com que o motor deve girar.;
- Se as voltagens forem iguais o motor deve parar.



III. RESULTADOS

O circuito obtido após a montagem do esquemático está representado na figura 5. A estrutura proposta na figura 6 simula um rastreador.

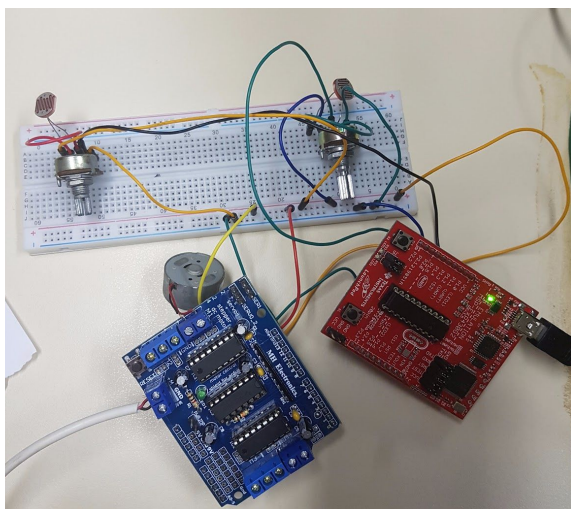


Figura 5: Circuito obtido

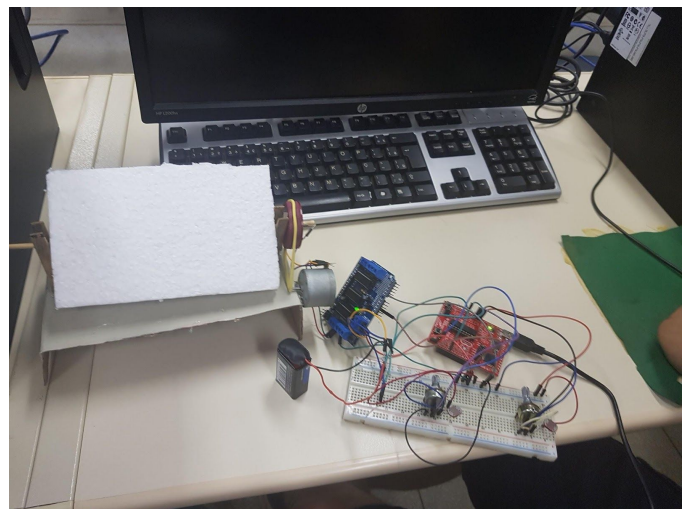


Figura 6: Estrutura do rastreador

Foi possível perceber que os leds destinados a indicar a presença de luz acendem corretamente quando colocados sob ela. O que representa que os sensores estão se comunicando corretamente com o microprocessador.

No que diz respeito ao motor e seu respectivo driver (ponte H) observou-se que ele não se comportou da maneira adequada, o sensor direito está funcionando de maneira inversa a proposta no projeto, ou seja, quando existe a presença de luz ele não rotaciona e no caso contrário rotaciona. A velocidade de rotação do motor foi adequada a proposta do projeto e o problema de instabilidade devido a calibração foi solucionado. Por fim vale ressaltar que o fato do rastreador estar funcionando apenas para um lado é muito provavelmente devido a tensão de alimentação provida do MSP430 ser de 3.3 V o que seria uma tensão muito baixa e que funcionaria melhor se alimentada a 5 V.

IV. REFERÊNCIAS

- [1] “Energia solar”, um breve resumo”, Aneel. Acesso em 04/09/2017. Disponível em: [http://www2.aneel.gov.br/aplicacoes/atlas/pdf/03-energia_solar\(3\).pdf](http://www2.aneel.gov.br/aplicacoes/atlas/pdf/03-energia_solar(3).pdf)
- [2] Princeton University. «[Photoresistor](#)». Consultado em 8 de Outubro de 2017

APÊNDICE

```
#INCLUDE "msp430.h"

#define ADC_CHANNELS 2

UNSIGNED INT SAMPLES[ADC_CHANNELS];

#define LED1 BIT4

#define LED2 BIT6

#define SENSOR_LEFT BIT0

#define SENSOR_GND BIT2

#define SENSOR_RIGHT BIT1

#define SENSOR_GND1 BIT3

#define RED_LED LED1

#define GRN_LED LED2

VOID CONFIGUREADC(VOID) {

ADC10CTL1 = INCH_1 | ADC10DIV_0 | CONSEQ_3 | SHS_0;

ADC10CTL0 = SREF_0 | ADC10SHT_2 | MSC | ADC10ON | ADC10IE;

ADC10AE0 =SENSOR_LEFT + SENSOR_RIGHT ;

ADC10DTC1 = ADC_CHANNELS;

}

VOID MAIN(VOID) {

WDTCTL = WDTPW | WDTHOLD;

BCSCTL1 = CALBC1_1MHZ;

DCOCTL = CALDCO_1MHZ;

BCSCTL2 &= ~(DIVS_3);

P1DIR = 0; /* SET AS INPUTS */

P1SEL = 0; /* SET AS DIGITAL I/Os */

P1OUT = 0; /* SET RESISTORS AS PULL-DOWNS */

P1REN = 0xFF; /* ENABLE PULL-DOWN RESISTORS */

P2DIR = 0; /* SET AS INPUTS */

P2SEL = 0; /* SET AS DIGITAL I/Os */

P2OUT = 0; /* SET RESISTORS AS PULL-DOWNS */

P2REN = 0xFF; /* ENABLE PULL-DOWN RESISTORS */
```

```

P1REN &= ~(LED1 | LED2); /* DISABLE PULL-UP/DOWNS */

P1DIR |= (LED1 | LED2); /* CONFIGURE AS OUTUTS */

P1REN &= ~(SENSOR_GND | SENSOR_GND1); /* DISABLE PULL-UP/DOWN */

P1OUT &= ~(SENSOR_GND | SENSOR_GND1); /* SENSOR_GND SHOULD BE AT GND */

P1DIR |= (SENSOR_GND | SENSOR_GND1); /* SENSOR_GND MUST BE AN OUTPUT */

P1REN |= (SENSOR_LEFT | SENSOR_RIGHT); /* ENABLE PULL-UP ON SENSOR */

P1IN |= (SENSOR_LEFT | SENSOR_RIGHT); /* SET RESISTOR AS PULL-UP */

CONFIGURE_ADC();

__ENABLE_INTERRUPT();

while (1) {

    __DELAY_CYCLES(1000);

    ADC10CTL0 &= ~ENC;

    while (ADC10CTL1 & BUSY);

    ADC10SA = (UNSIGNED_INT) SAMPLES;

    ADC10CTL0 |= ENC + ADC10SC;

    __BIS_SR_REGISTER(CPUOFF + GIE);

    if (SAMPLES[0] < SAMPLES[1]) {

        P1OUT |= RED_LED;

        P1OUT &= ~(GRN_LED);

    } else if (SAMPLES[0] == SAMPLES[1]) {

        P1OUT &= ~(RED_LED);

        P1OUT &= ~(GRN_LED);

    } else {

        P1OUT |= GRN_LED;

        P1OUT &= ~(RED_LED);

    }

}

#pragma vector=ADC10_VECTOR

__interrupt void ADC10_ISR (void) {

    __BIC_SR_REGISTER_ON_EXIT(CPUOFF);

}

```

