



FACULDADE SENAI DE TECNOLOGIA MECATRÔNICA

PROJETO GIRASSOL ADEWALE

ADEWALE SUN TRACKER

Adewale Olusegun Ige ^{1 i}

Paulo Sebastião Ladivez ^{2 ii}

José Roberto dos Santos ^{3 iii}

Nelson Wilson Paschoalinoto ^{4 iv}

RESUMO

A energia solar é um tipo de energia renovável e limpa, amplamente disponível em nosso planeta. A transformação deste tipo de energia em energia elétrica aumenta a cada dia, como forma de atender a demanda pelo uso e reduzir o uso de outras fontes de energia, que causam algum tipo de dano ao nosso planeta. Como a vida moderna precisa de energia elétrica para atender às suas necessidades diárias, sejam essas domésticas ou industriais, o desenvolvimento deste trabalho tem como base um projeto de transformação de energia solar em elétrica com um sistema de rastreamento da energia solar, sistema este que é capaz de posicionar suas placas fotovoltaicas em função do movimento de rotação e translação da Terra, propiciando um aumento da eficiência.

ABSTRACT

Solar energy is a type of renewable and clean energy widely available on our planet. The transformation of this type of energy into electricity increases every day as a way to meet the demand for use and to reduce the use of other energy sources that cause some damage to our planet. As the modern life needs electricity to meet its daily needs, whether domestic or industrial, this work is based on the development of a project to transform solar energy into electricity with solar energy tracking system, it is able to position its photovoltaic plates as a function of the Earth's rotational and translation motion, providing an increase in efficiency.

1 INTRODUÇÃO

A energia solar pode ser convertida em várias formas de energia, dependendo da tecnologia que se usa. Entre essas formas de energia está a transformação em energia elétrica

¹ Pós-graduando em Automação Industrial na Faculdade SENAI de Tecnologia Mecatrônica. E-mail: victorpowergen@gmail.com

² Professor da Faculdade SENAI de Tecnologia Mecatrônica. E-mail: paulo.ladivez@sp.senai.br

³ Professor da Faculdade SENAI de Tecnologia Mecatrônica. E-mail: joseroberto @sp.senai.br

⁴ Professor da Faculdade SENAI de Tecnologia Mecatrônica. E-mail: nelson.paschoalinoto@sp.senai.br

por meio da utilização de células fotovoltaicas de policloreto de vinila (PVC), também conhecidas como painéis solares, que são capazes de converter energia solar diretamente em eletricidade.

Este projeto tem como base o desenvolvimento de um sistema manipulador capaz de alterar o posicionamento das células fotovoltaicas, de maneira que estas possam acompanhar a incidência dos raios solares desde o nascer até o pôr do sol, aumentando a eficiência na transformação da energia, quando comparado com os sistemas convencionais de célula fotovoltaica com posicionamento fixo.

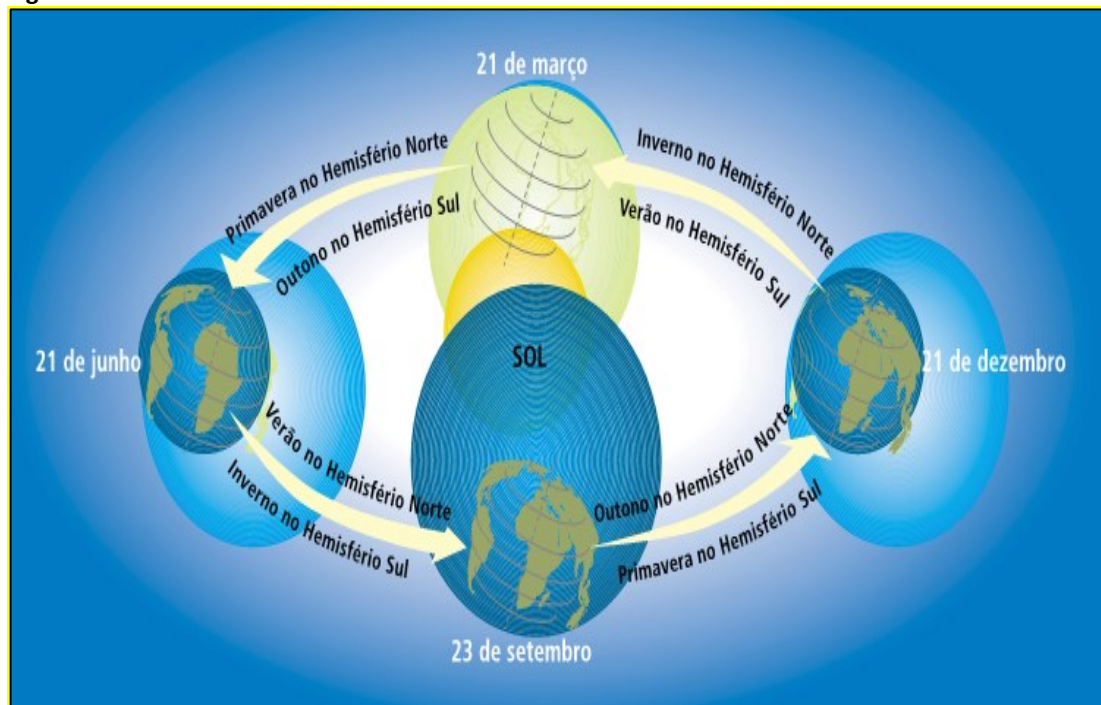
2 DESENVOLVIMENTO

Os sistemas de energia solar fotovoltaicos são sistemas capazes de gerar energia elétrica, por meio da incidência de luz solar recebida sobre placas. Quando ocorre uma diminuição da incidência dos raios solares, seja esta provocada por objetos, edificações, árvores ou outro fator, o fluxo de elétrons que circula pelo módulo solar é diminuído e consequentemente, a produção de energia elétrica é reduzida.

Para maximizar o aproveitamento da radiação solar em função do movimento de rotação e translação da Terra, faz-se necessário o ajuste adequado do posicionamento do painel solar fotovoltaico para que possa acompanhar a trajetória do sol.

A figura 1 a seguir, ilustra as representações das estações do ano e o movimento da Terra em torno do Sol.

Figura 1 – Movimento da Terra em torno do Sol



Fonte: MAGNOLI; SCALZARETTO, 1998 apud ANEEL.

No desenvolvimento deste projeto, será utilizada uma placa Arduino Mega, sensores e atuadores pertencentes a esta plataforma, de modo a garantir o melhor posicionamento da incidência de luz solar sobre o painel fotovoltaico.

Segundo McRoberts (2011, p. 22), “um Arduino é um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e componentes externos conectados a ele”.

Monk (2013, p. 5) define o Arduino como uma “plataforma de microcontrolador que atraiu a imaginação dos entusiastas de eletrônica”, por ser de fácil utilização.

Com base nas informações descritas anteriormente sobre maximizar o aproveitamento da incidência dos raios solares e uma prévia sobre as definições do que é a plataforma Arduino, foi desenvolvido um sistema de rastreamento solar, utilizando o Arduino Mega para processar o efeito de rotação e de translação da Terra. Sensores e atuadores posicionarão a célula fotovoltaica, objetivando a melhor condição de aproveitamento de transformação de energia.

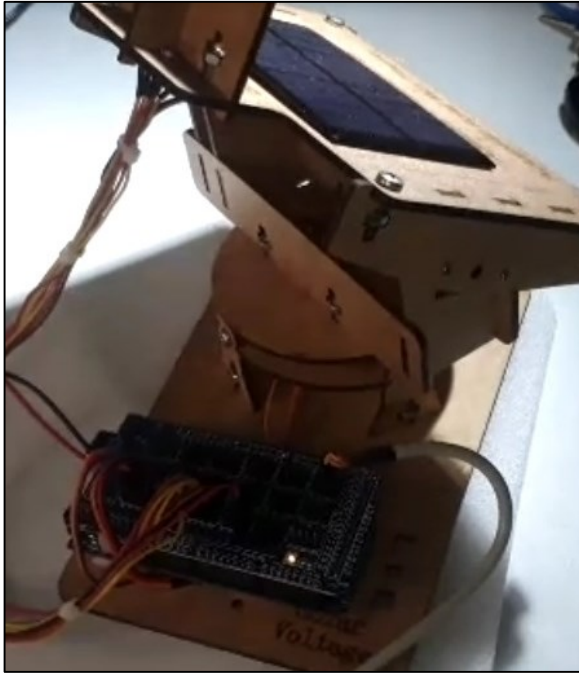
2.1 Descrição do Processo

O sistema de rastreamento solar consiste na utilização de um microcontrolador ATmega 2560, célula fotovoltaica, resistores do tipo *Light Dependent Resistor* (LDR), um *display 16X2 Liquid Crystal Display* (LCD), *hardware* para servo motor conhecido como *shield*, servomotor para posicionamento da célula fotovoltaica e suportes para fixação e posicionamento dos componentes citados anteriormente.

Os resistores LDR são componentes que alteram o seu valor de resistência elétrica dependendo da quantidade de luz recebida por ele. Desta forma foram utilizados no projeto para detectar a incidência solar, enviando essa informação ao Arduino Mega 2560, e que por sua vez irá ajustar a posição das células fotovoltaicas através dos servos motores, desta maneira as placas fotovoltaicas sempre ficarão posicionadas de maneira a se obter o melhor aproveitamento da energia solar.

A figura 2 ilustra o protótipo do sistema de rastreamento solar, detalhado no decorrer deste trabalho.

Figura 2 – Protótipo do sistema de rastreamento solar



Fonte: Elaborado pelo autor

Os quadros 1 e 2 demonstram os comandos da programação desenvolvidos e embarcados no microcontrolador, através do Ambiente de Desenvolvimento Integrado do Arduino (IDE), para o Arduino ATmega 2560 e para o nodeMCU.

Quadro 1 – Comandos para o Arduino ATmega 2560

```
#include <Servo.h>           //Biblioteca dos Servos
#include <LiquidCrystal_PCF8574.h> //Biblioteca do LCD

//Defines
#define ldrTL A0
#define ldrTR A1
#define ldrBL A2
#define ldrBR A3
#define PL_SOLAR A6

int posH = 90;           // posição inicial do servo
int posV = 45;           // posição inicial do servo
int qtdeAmostras = 20;   // Qtde de amostras lidas nos sensores
unsigned long tempo_amostragem = 0; // tempo de referência das leituras dos sensores LDR
int tp = 200;            // intervalo entre as leituras dos sensores LDR
float ps = 0.0;          //Guarda a tensão da Placa Solar

//Caracter ° para o LCD
byte customChar[8] = {    // Matriz do caracter especial (°)
  0b01100,
  0b10010,
  0b10010,
  0b01100,
  0b00000,
  0b00000,
  0b00000,
  0b00000,
```

```

0b00000
};
//Instancia os objetos com a classe Servo
Servo ServoHorizontal;
Servo ServoVertical;

//Instancia o objeto lcd com a classe LiquidCrystal_PCF8574
LiquidCrystal_PCF8574 lcd(0x27); // set the LCD address to 0x27 for a 16 chars and 2 line display

void setup() {
  Serial2.begin(9600); //inicializa a Serial2
  //Configura o modo de operação das portas
  pinMode(ldrTL, INPUT);
  pinMode(ldrTR, INPUT);
  pinMode(ldrBL, INPUT);
  pinMode(ldrBR, INPUT);
  pinMode(46, OUTPUT);
  pinMode(47, OUTPUT);

  //Inicializa os Servos
  ServoHorizontal.attach(46);
  ServoVertical.attach(47);

  //Coloca os Servos nas posições iniciais (90 e 45 graus)
  ServoVertical.write(posV);
  delay(20);
  ServoHorizontal.write(posH);
  delay(1000);

  //Inicializa o LCD
  lcd.begin(16, 2);
  lcd.createChar(1, customChar);
  lcd.setBacklight(255);
  lcd.clear();

  lcd.setCursor(0, 0);
  lcd.print("PROJETO GIRASSOL");

  lcd.setCursor(0, 1);
  lcd.print("*** ADEWALE ***");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("H:   V:   ");
  lcd.setCursor(0, 1);
  lcd.print("TENSAO: ");
  lcd.setCursor(3, 0);
  lcd.print("   ");
  lcd.setCursor(3, 0);
  lcd.print(ServoHorizontal.read());
  lcd.write((byte)1);
  lcd.setCursor(11, 0);
  lcd.print("   ");
  lcd.setCursor(11, 0);
  lcd.print(ServoVertical.read());

```

```

lcd.write((byte)1);
}

void loop() {
  if (tempo_amostragem < millis()) {
    long tl = 0;
    long tr = 0;
    long bl = 0;
    long br = 0;
    ps = analogRead(PL_SOLAR) * 5.0 / 1023 ; //valor em volts medido da Placa Solar
    for (int i = 0; i < qtdeAmostras; i++) {
      tl += analogRead(ldrTL);
      tr += analogRead(ldrTR);
      bl += analogRead(ldrBL);
      br += analogRead(ldrBR);
    }
    //calcula da media
    tl = tl / qtdeAmostras;
    tr = tr / qtdeAmostras;
    bl = bl / qtdeAmostras;
    br = br / qtdeAmostras;

    //offset para equilibrar os valores lidos nos sensores LDR
    tr = tr - 28;
    bl = bl + 50;
    br = br - 25;
    //mostra os valores lidos no LCD
    lcd.setCursor(3, 0);
    lcd.print(" ");
    lcd.setCursor(3, 0);
    lcd.print(posH);
    lcd.write((byte)1);

    lcd.setCursor(11, 0);
    lcd.print(" ");
    lcd.setCursor(11, 0);
    lcd.print(posV);
    lcd.write((byte)1);

    lcd.setCursor(11, 1);
    lcd.print(" ");
    lcd.setCursor(11, 1);
    lcd.print(String(ps, 1));
    lcd.print("V");

    sendEsp8266(); //envia os dados lidos para o esp8266 via serial

    //Atualiza o tempo de amostragem
    tempo_amostragem = millis() + tp;

    //movimento horizontal
    if ( bl > br + 30) {
      if (posH > 10) {
        posH--;
        ServoHorizontal.write(posH);

```

```

    delay(20);
    return;
  }
}
if ( bl < br ) {
  if (posH < 170) {
    posH++;
    ServoHorizontal.write(posH);
    delay(20);
    return;
  }
}
//movimento vertical
if (tl < bl) {
  if (posV > 10) {
    posV--;
    ServoVertical.write(posV);
    delay(20);
    return;
  }
  if (tl > bl + 50) {
    if (posV < 120) {
      posV++;
      ServoVertical.write(posV);
      delay(20);
      return;
    }
  }
}

void sendEsp8266() {
  Serial2.println(String(posH) + "º," + String(posV) + "º," + String(ps) + "V,");
}

```

Fonte: Elaborado pelo autor

Quadro 2 – Comandos para o controlador nodeMCU

```

/* LIGAÇÃO SERIAL
  ARDUINO  RX 0 ----- D5 (TX) GPIO14 ESP8266
  ARDUINO  TX 1 ----- D6 (RX) GPIO12 ESP8266
*/

//BIBLIOTECAS=====
#include <ESP8266WiFi.h>      //biblioteca para wifi
#include <WiFiClient.h>       //biblioteca para wifi
#include <ESP8266WebServer.h>  //biblioteca para webserver
#include "index.h" //para javascripts
#include <SoftwareSerial.h>

//VARIÁVEIS=====
//SSID e Password usado no ESP Access Point
const char* ssid = "GIRASSOL";
const char* password = "123456789";
byte stSensor = 0; //status do sensor
int counter = 0; //contador de pecas

String tensao;

```

```

String pvertical;
String phorizontal;

String XML;

//OBJETOS=====
ESP8266WebServer server(80);    //cria o objeto server na porta 80
SoftwareSerial ESPSerial (12, 14); //emula uma porta serial virtual (rx,tx)
//CODIGO=====
void buildXML() {
  XML = "<?xml version='1.0'?>";
  XML += "<xml>";
  //cria as label XML atraves das TAG's criadas, podem possuir qualquer nome

  XML += "<t>";
  XML += tensao; //Exibe a tensão na pagina
  XML += "</t>";

  XML += "<h>";//
  XML += phorizontal; //Exibe a posição horizontal na pagina
  XML += "</h>";

  XML += "<v>";
  XML += pvertical; //Exibe a posição vertical na pagina
  XML += "</v>";

  XML += "</xml>";
}
//=====
// A rotina e executada ao acessar o IP no navegador
//=====
void handleRoot() {
  String s = MAIN_page; //le o conteúdo HTML
  server.send(200, "text/html", s); //envia a pagina
}
void handleXML() {
  buildXML();
  server.send(200, "text/xml", XML);
}
//=====
//      SETUP
//=====
void setup(void) {
  //Serial.begin(9600);
  ESPSerial.begin(9600); //serial que comunica com o Arduino Mega

  WiFi.mode(WIFI_AP); //inicializa a wifi em modo access point
  WiFi.softAP(ssid, password);

  delay(4000);

  //Webserver
  server.on("/", handleRoot); //rotina de inicialização
  server.on("/xml", handleXML);
  server.begin(); //inicializa o server

```



```

}
//=====
//          LOOP
//=====
void loop(void) {

    server.handleClient();

    //Captura as informações vindas da Serial (ESPSerial)
    int col = 0;
    String t = "";
    while (ESPSerial.available()) {
        char d = ESPSerial.read();
        t += d;
        delay(10);
    }

    //Separa as informações vindas da ESPSerial
    if (t != "") {
        phorizontal = getValue(t, ',', 0);
        pvertical = getValue(t, ',', 1);
        tensao = getValue(t, ',', 2);
    }

}

//Função para fazer separar a string
String getValue(String data, char separator, int index) {
    int found = 0;
    int strIndex[] = { 0, -1 };
    int maxIndex = data.length() - 1;

    for (int i = 0; i <= maxIndex && found <= index; i++) {
        if (data.charAt(i) == separator || i == maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ? i + 1 : i;
        }
    }
    return found > index ? data.substring(strIndex[0], strIndex[1]) : "";
}

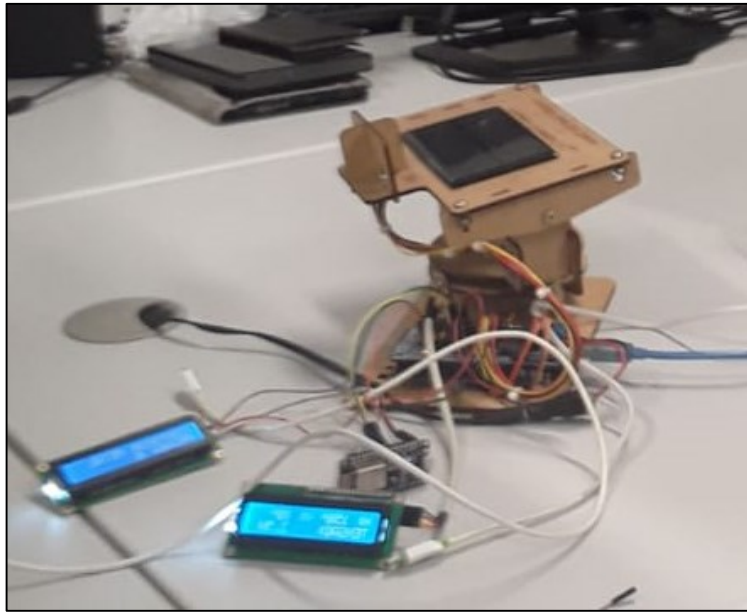
```

Fonte: Elaborado pelo autor

2.2 Descrição do projeto

Para o desenvolvimento deste projeto foi utilizado a linguagem de programação C++ criada para o Ambiente de Desenvolvimento Integrado ou *Integrated Development Environment* (IDE) do Arduino, que segundo McRoberts (2011) é um *software* livre no qual você escreve um programa de computador denominado de conjunto de instruções passo a passo, e ao fazer o *upload* (carregamento deste programa para o Arduino), este executará as instruções. A ilustração a seguir, figura 3 exibe testes preliminares realizados no protótipo.

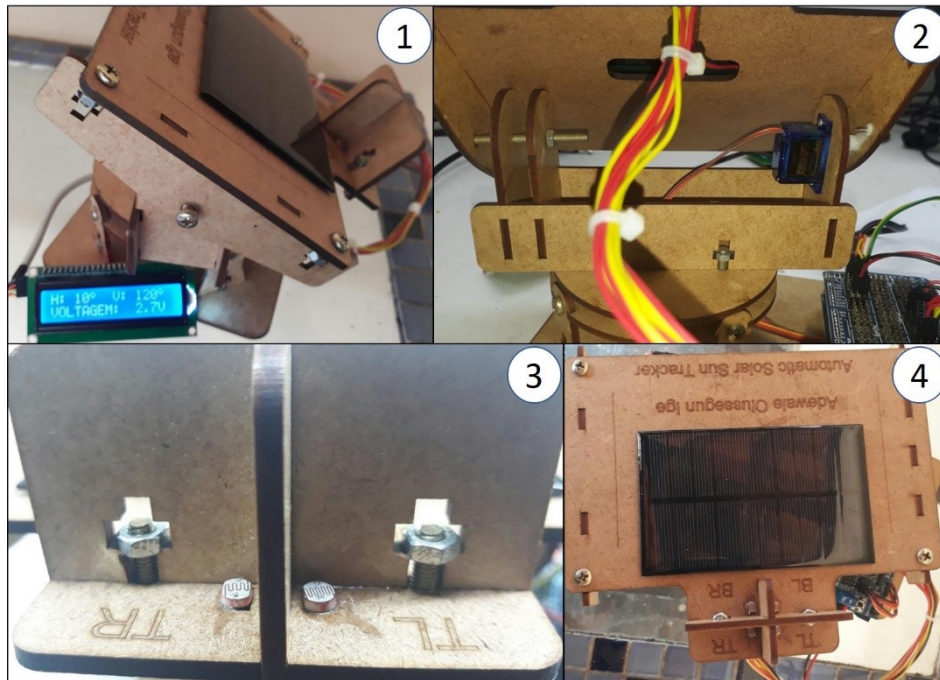
Figura 3 – Testes preliminares do sistema de rastreamento solar



Fonte: Elaborado pelo autor

Na figura 4, é possível observar os detalhes do projeto, como por exemplo o *display* de LCD, que informa o posicionamento horizontal e vertical da placa fotovoltaica (1), um dos servomotores, que ajusta o posicionamento da placa fotovoltaica (2), os resistores do tipo LDR, que informam ao Arduino a incidência solar (3) e por último uma visão geral da placa fotovoltaica e os resistores do tipo LDR (4).

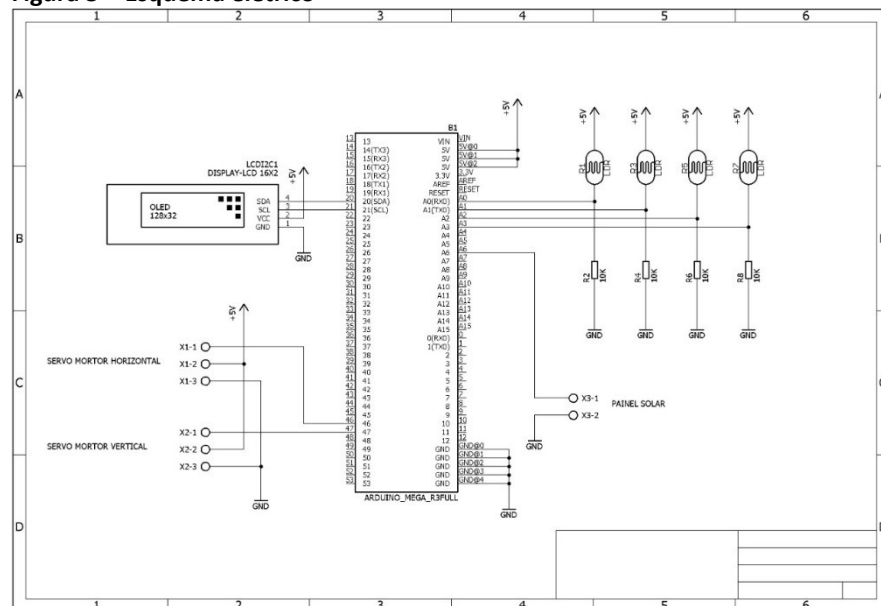
Figura 4 – Detalhe da montagem do projeto



Fonte: Elaborado pelo autor

Na figura 5 observa-se o esquema elétrico utilizado no desenvolvimento do projeto.

Figura 5 – Esquema elétrico



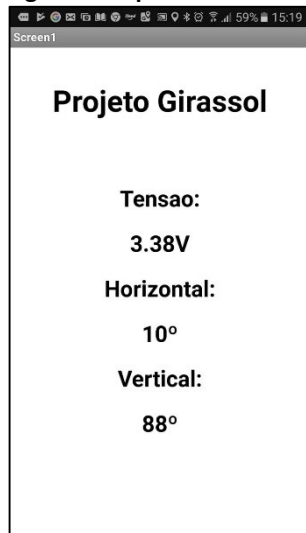
Fonte: Elaborado pelo autor

Dando continuidade ao projeto, um *hardware* do modelo nodeMCU foi implementado ao sistema de rastreamento solar, de maneira a permitir que os valores de tensão e posicionamento da placa fotovoltaica pudessem ser observados de maneira remota. Visando

viabilizar esse recurso, foi necessário o desenvolvimento de um aplicativo de celular na plataforma gratuita denominada MIT App Inventor.

A figura 6 ilustra o resultado da aplicação no celular com os respectivos valores de tensão e posicionamento da placa.

Figura 6 – Aplicativo de celular



Fonte: Elaborado pelo autor

3 CONCLUSÃO

A busca por fontes de energias renováveis tem aumentado exponencialmente nos últimos anos, visto que esta garante a sustentabilidade do planeta.

Embasado na ideia de energias renováveis, este trabalho foi realizado para maximizar a energia disponibilizada pelo Sol e demonstrar que é possível realizar o rastreamento da incidência de luz solar, devido ao movimento realizado pelo Sol de leste a oeste.

Os sistemas de rastreamento de energia solar de dois eixos são capazes de aumentar a transformação de energia solar em energia elétrica, quando comparado com sistemas de energia solar com placas fixas, por assim garantirem o melhor posicionamento do painel fotovoltaico a qualquer hora do dia de acordo com o movimento do Sol, reduzindo também o número de placas utilizadas em um projeto e consequentemente o espaço necessário para a acomodação das placas.

REFERÊNCIAS

MAGNOLI, Demétrio; SCALZARETTO, Reinaldo. Geografia, espaço, cultura e cidadania. v.1. São Paulo: Moderna, 1998. In: **Energia solar**. Agência Nacional de Energia Elétrica - ANEEL. Disponível em: http://www2.aneel.gov.br/aplicacoes/atlas/energia_solar/3_2.htm. Acesso em: 30 jun. 2020.

MCROBERTS, Michael. **Arduino básico**. São Paulo: Novatec, 2011. 456 p.

MONK, Simon. **Programação com Arduino: começando com SKETCHES**. 1 ed. Porto Alegre: Bookman, 2013, 147 p.

AGRADECIMENTOS

Agradeço a minha esposa Aderonke Esther Ige e minhas filhas Grace, Favour, Mercy por compreenderem as horas ausentes para que fosse possível a conclusão deste trabalho.

Aos amigos Daniel Soares de Oliveira, Alexandre Benedetti que contribuíram na execução desse projeto.

A todo o corpo docente, direção e administração desta Faculdade Escola SENAI Armado de Arruda Pereira, que buscam incansavelmente transpor conhecimentos para com os alunos de forma clara e objetiva.

Agradeço especialmente a meu professor orientador Paulo Sebastião Ladivez, por aceitar a ideia deste trabalho e me orientar de maneira clara e objetiva para que este fosse concluído.

Sobre os autores:

ⁱ Adewale Olusegun Ige



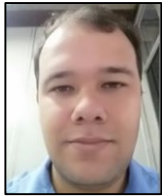
Possui graduação em Tecnologia de Petróleo e Gás pelas Faculdades Metropolitanas Unidas-FMU. São Paulo (2015). Terminou os créditos do curso de pós-graduação em Automação Industrial em dezembro de 2017 pela Escola SENAI Armando de Arruda Pereira. Presidente da empresa Victor Powergen Energia Renovável.

ⁱⁱ Paulo Sebastião Ladivez



Possui graduação em Engenharia Elétrica pela Universidade Mogi das Cruzes (1984) com especialização em Tecnologias e Sistemas de Informação pela Universidade Federal do ABC (2013). Atualmente é professor da Faculdade SENAI de Tecnologia Mecatrônica, lecionando as disciplinas Projetos Mecatrônicos, Microcontroladores no Curso Tecnólogo em Mecatrônica Industrial e na Pós-Graduação em Automação Industrial. Tem experiência na área de Engenharia Eletrônica, com ênfase em Automação Industrial e Mecatrônica, atuando principalmente nos seguintes temas: Mecatrônica, Manufatura Digital, Redes Industriais, Automação Industrial, Microcontroladores e Controle.

iii JOSÉ ROBERTO DOS SANTOS



Atualmente ministra aulas na pós-graduação de Indústria 4.0 e na graduação em Tecnologia em Mecatrônica na Faculdade SENAI de Tecnologia Mecatrônica, que fica no SENAI Armando de Arruda Pereira. Assessora também o Instituto SENAI de Tecnologia Metalmeccânica em projetos industriais com foco na Indústria 4.0. Durante 9 anos ministrou aulas pelo SENAI-SP, nos cursos de técnico em eletroeletrônica, cursos de aprendizagem industrial eletricitista de manutenção e mecânico de usinagem, além de Formação Inicial e Continuada (FIC) com cursos voltados a área de redes de computadores e programação, possui treinamento de Linux, cisco e Microsoft. Possui Pós-graduação na área de segurança da informação pela Uninove (2016), graduação em tecnologia da informação e bacharel em sistema da informação (2009), além de superior em Automação industrial. Tem experiência na área de Segurança da informação, administração de ambientes de redes Windows e Linux, automação indústria.

iv VICENTE GOMES DE OLIVEIRA JUNIOR



Professor. Mestre em Engenharia Mecânica com ênfase em Materiais e Processos. É pós-graduado em Engenharia de Processos Industriais - Instrumentação, Automação e Controle. Possui graduação em Tecnologia Mecânica - Processos de Produção e Licenciatura Plena em Mecânica. Atualmente é Professor da Faculdade SENAI de Tecnologia Mecatrônica. Tem experiência na área de Engenharia Mecânica, com ênfase em Processos de Fabricação, CAD e em Coordenação Pedagógica no ensino técnico.