

Global Solution - 1o Semestre

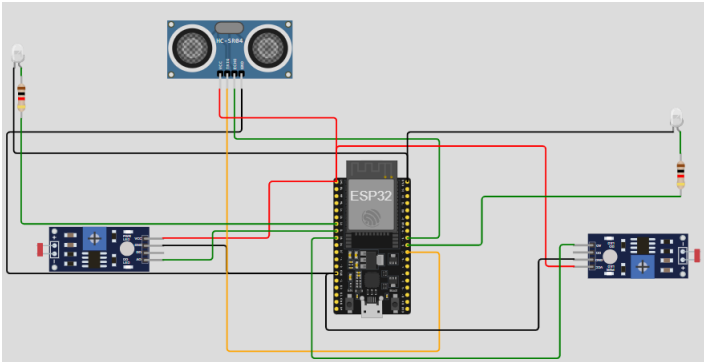
Introdução

Dentro do tema geral da Global Solution desse semestre, a Inteligencia Artificial tem como missão de desenvolver uma solução baseada em Data Science, IoT, Python e Banco de Dados que auxiliem na otimização do consumo de energia em ambientes residenciais, comerciais ou urbanos.

O projeto vem sendo desenvolvido como foco na melhoria da eficiência energética, integrando fontes renováveis e considerando dados em tempo real para tomar decisões automáticas que promovam a sustentabilidade.

Para desenvolver essa solução, dentre as disciplinas a serem exploradas temos o tema de *Artificial Intelligence with Computer Systems and Sensors (AICSS)*. O objetivo nesta parte é desenvolver um circuito com o ESP32 que otimize a iluminação interna e externa de uma residencia.

Descrição da Solução



Conforme representado na figura acima, temos um projeto desenvolvido com o ESP32 como peça dentral. Separamos o projeto em duas partes para facilitar o entendimento: um para ambiente externo (lado direito) e um para ambiente interno (lado esquerdo).Foram adicionados dois sensores LDR para medição de luminosidade, uma para cada ambiente, assim como dois LED's que buscam representar a iluminação de cada ambiente respectivamente. Adicionalmente, foi incluído para o ambiente interno um sensor HC-SR04 que faz medição de distancias, onde ele foi utilizado como um sensor mais eficaz para detecção de presença.

O código então foi desenvolvido para representar uma logica em que:

- 1. Ambiente Interno:** Caso tenhamos boa luminosidade no ambiente intertno, a luz não será acionada. Entretanto ao escurecer a luz somente será acesa se for detectado algum movimento. Para isso o sensor HC-SR04 está configurado como default para 400 cm. Caso seja detectado uma diminuição dessa distancia para menos de 200 cm (quando alguem passar na frente do sensor por exemplo), o sistema então libera para que a luz seja acesa caso a luminosidade identificada tenha um valor analogico maior que 100 (algo em terno de 2.000 LUX - podemos ajustar a sensibilidade do sensor LDR como desejado);
- 2. Ambiente Externo:** Para o ambiente externo não utilizamos alguma técnica de detecção de movimento pois a luz precisa estar sempre acessa quando a luminosidade estiver baixa por questões de segurança que usam câmeras sem infravermelho no local. Neste caso o sistema ascende a luz caso a luminosidade identificada tenha um valor analogico maior que 100 (algo em terno de 2.000 LUX - podemos ajustar a sensibilidade do sensor LDR como desejado);

Código Fonte

Projeto.ino

```
int LDRSensor_Interno = 33; // Pino do sensor de luminosidade interno
int LDRSensor_Externo = 25; // Pino do sensor de luminosidade externo
int Luz_Interna = 32; // Pino da luz interna
int Luz_Externa = 5; // Pino da luz externa
int Trig_pin = 17; // Pino TRIG do Sensor HC-SR04
int Echo_pin = 18; // Pino ECHO do Sensor HC-SR04

// Variáveis para cálculo de distância do Sensor HC-SR04
long duration;
float Speed_of_sound = 0.034;
float dist_in_cm;

void setup() {
  Serial.begin(115200);
  pinMode(LDRSensor_Interno, INPUT);
  pinMode(LDRSensor_Externo, INPUT);
  pinMode(Luz_Interna, OUTPUT);
  pinMode(Luz_Externa, OUTPUT);
  pinMode(Trig_pin, OUTPUT);
  pinMode(Echo_pin, INPUT);
}

void loop() {
  // Leitura dos sensores de luminosidade
  int luminosidadeInterna = analogRead(LDRSensor_Interno);
  Serial.println("LUX Analog Read Interno: " + String(luminosidadeInterna));

  int luminosidadeExterna = analogRead(LDRSensor_Externo);
  Serial.println("LUX Analog Read Externo: " + String(luminosidadeExterna));

  // Leitura do sensor de distância HC-SR04
  digitalWrite(Trig_pin, LOW);
  delayMicroseconds(2);
  digitalWrite(Trig_pin, HIGH);
  delayMicroseconds(10);
  digitalWrite(Trig_pin, LOW);
  duration = pulseIn(Echo_pin, HIGH);
  dist_in_cm = duration * Speed_of_sound / 2;
  Serial.println("Distancia em cm: " + String(dist_in_cm));

  //Tabela de Parametros do Sensor de Luminosidade
  //-----
  //Condição          Nivel de iluminância (lux)    analogRead()
  //Luz cheia          0.1                          1016
  //Crepúsculo profundo 1                          985
  //Crepúsculo         10                         853
  //Monitor do computador** 50                     633
  //Iluminação da escada 100                       511
```

```

//Iluminação do escritório 400          281
//Dia nublado              1,000        170
//Luz do dia               10,000       39
//Luz direta do sol        100,000      8

// Controle da luz externa
if (luminosidadeExterna > 100) {
  digitalWrite(Luz_Externa, HIGH); // Liga a luz externa
} else {
  digitalWrite(Luz_Externa, LOW); // Desliga a luz externa
}

// Controle da luz interna
if (luminosidadeInterna > 100 && dist_in_cm < 200) {
  digitalWrite(Luz_Interna, HIGH); // Liga a luz interna
  delay(10000);
} else {
  digitalWrite(Luz_Interna, LOW); // Desliga a luz interna
}

delay(500);
}

```

▼ Diagrama JSON

```

{
  "version": 1,
  "author": "André Cunha",
  "editor": "wokwi",
  "parts": [
    {
      "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },
    {
      "type": "wokwi-photoresistor-sensor",
      "id": "ldr1",
      "top": 89.6,
      "left": -402.4,
      "attrs": {}
    },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -166.8,
      "left": -447.4,
      "attrs": { "color": "white", "lightColor": "orange" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r1",
      "top": -91.2,
      "left": -451.75,
      "rotate": 90,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-hc-sr04",
      "id": "ultrasonic1",
      "top": -190.5,
      "left": -224.9,
      "attrs": { "distance": "400" }
    },
    {
      "type": "wokwi-photoresistor-sensor",
      "id": "ldr2",
      "top": 99,
      "left": 324.4,
      "rotate": 180,
      "attrs": {}
    },
    {
      "type": "wokwi-led",
      "id": "led2",
      "top": -80.4,
      "left": 445.4,
      "attrs": { "color": "white", "lightColor": "orange" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r2",
      "top": 14.4,
      "left": 441.05,
      "rotate": 90,
      "attrs": { "value": "1000" }
    }
  ],
  "connections": [
    [ "esp:TX", "$serialMonitor:RX", "", [] ],
    [ "esp:RX", "$serialMonitor:TX", "", [] ],
    [ "ldr1:VCC", "esp:3V3", "red", [ "h144", "v-76.8" ] ],
    [ "ldr1:GND", "esp:GND.1", "black", [ "h86.4", "v38" ] ],
    [ "ldr1:A0", "esp:33", "green", [ "h67.2", "v172.1" ] ],
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "led1:A", "r1:1", "green", [ "v0" ] ],
    [ "esp:GND.2", "led1:C", "black", [ "v-38.4", "h-437.16" ] ],
    [ "ultrasonic1:GND", "esp:GND.1", "black", [ "v57.6", "h-318", "v192" ] ],
    [ "ultrasonic1:ECHO", "esp:18", "green", [ "v67.2", "h277.6", "v134.4" ] ],
    [ "ultrasonic1:TRIG", "esp:17", "orange", [ "v355.2", "h287.6", "v-134.4" ] ],
    [ "ultrasonic1:VCC", "esp:3V3", "red", [ "v38.4", "h-110.59" ] ],
    [ "ldr2:VCC", "esp:3V3", "red", [ "h-38.4", "v-163.2", "h-283.39" ] ],
    [ "ldr2:GND", "esp:GND.1", "black", [ "h-48", "v96.4", "h-288", "v-76.8" ] ],
    [ "ldr2:A0", "esp:25", "green", [ "h-19.2", "v154.3", "h-336", "v-163.2" ] ],
    [ "r2:1", "led2:A", "green", [ "h0" ] ],
    [ "esp:GND.2", "led2:C", "black", [ "v0" ] ],
    [ "r2:2", "esp:5", "green", [ "h-268.8", "v66" ] ],
    [ "r1:2", "esp:32", "green", [ "v85.2", "h417.41" ] ]
  ],
  "dependencies": {}
}

```

Link de Video Explicativo postado no YouTube

Para uma demonstração prática da solução no Wokwi, veja: <https://www.youtube.com/watch?v=On0LvKlGCsQ>

O projeto também pode ser acessado em: <https://wokwi.com/projects/414376808765722625>

▼ Conclusão

A solução desenvolvida cumpriu exatamente seu propósito ao otimizar o consumo de energia de maneira conciente e efetivamente ascendendo somente se estritamente necessário conforme as regras definidas.

Notou-se um ponto de melhoria do código ao trazer a regra da iluminação externa para ser executada antes da iluminação interna uma vez que quando a iluminação interna é acionada temos um delay programado e isso acabava impedindo a atualização mais dinamica da iluminação externa. Após esse ajuste o código trouxe ainda mais eficiente energética.