

Sensor de Luminosidade- LDR

ESTUDO E TESTE DAS SUAS CAPACIDADES

Nelson Proença | LDR | November 18, 2019

Conteúdo

Sensor de luminosidade (LDR: ***L**ight **D**ependent **R**esistor*) – **fotorresistência** 2
Estrutura do Sensor 2
Esquema do Sensor 3
Aplicação do LDR 4
Exemplo de teste do Sensor 4
Material usado 4
Circuito utilizado 5
Código usado 6
Explicação do código 7
Conclusão 8

Sensor de luminosidade (LDR: *Light Dependent Resistor*) – fotorresistência

Estrutura do Sensor

O sensor de luminosidade, **resistor dependente de luz (BR)** ou **fotorresistência(EU)**, é conhecido pela sigla inglesa **LDR** (***Light Dependent Resistor***), é um componente eletrônico passivo do tipo resistência variável, ou seja, resistência que varia conforme a intensidade da luz que incide sobre ele.

Á medida que a intensidade da luz aumenta, a sua resistência diminui. O LDR 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 libertam elétrons para a banda condutora que irão melhorar a sua condutividade e assim diminuir a resistência.

Através de um multímetro podemos encontrar a resistência na escuridão ou na presença de luz intensa.

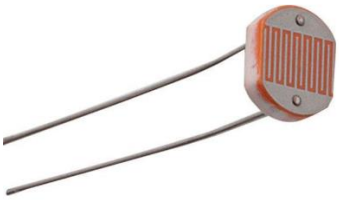
Os resultados típicos para um LDR poderão ser:

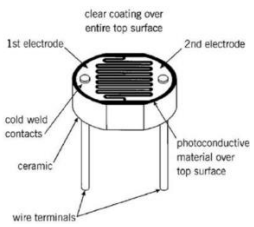
Escuridão: resistência máxima - mega ohms.

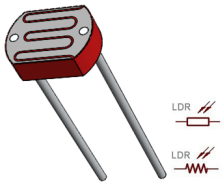
Muita Luz: resistência mínima - dezenas de ohms.

Dependendo do tipo, um LDR pode ser sensível às faixas de luz: Infravermelhos(IR), Luz visível ou Ultravioletas (UV).

O Sensor LDR:







Esquema do Sensor

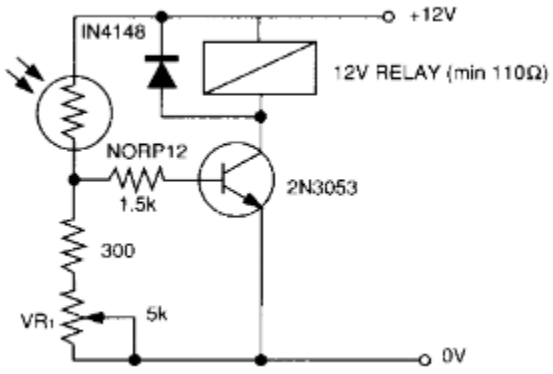
Características da memória de luz

Os resistores dependentes de luz têm uma propriedade específica, que lembram as condições de iluminação em que foram armazenados. Esse efeito de memória pode ser bem-limitado, armazenando os LDRs à luz antes do uso. O armazenamento leve reduz o tempo de equilíbrio para atingir valores de resistência constantes.

Classificações máximas absolutas:

- Pico de tensão, CA ou CC: 100V
- Corrente: 5mA
- Dissipação de energia em 25°C: 50mW
- Faixa de temperatura operacional: -25°C +75°C

Parameter	Conditions	Min.	Typ.	Max.	Units
Cell resistance	1000 lux 10 lux	- -	400 9	- -	Ω kΩ
Dark resistance	-	1.0	-	-	MΩ
Dark capacitance	-	-	3.5	-	pF
Rise time 1	1000 lux 10 lux	- -	2.8 18	- -	ms ms
Fall time 2	1000 lux 10 lux	- -	48 120	- -	ms ms



Exemplo de esquema para um LDR.

Aplicação do LDR

Este sensor poder ser aplicado em diversos equipamentos eletrónicos:

- Luz de rua automática
- Detetar dia ou noite
- Dimmer principal automático da luz
- Sensor de posição
- Usado junto com o LED como detetor de obstáculos
- Luzes automáticas do quarto
- Espelho retrovisor automático
- …

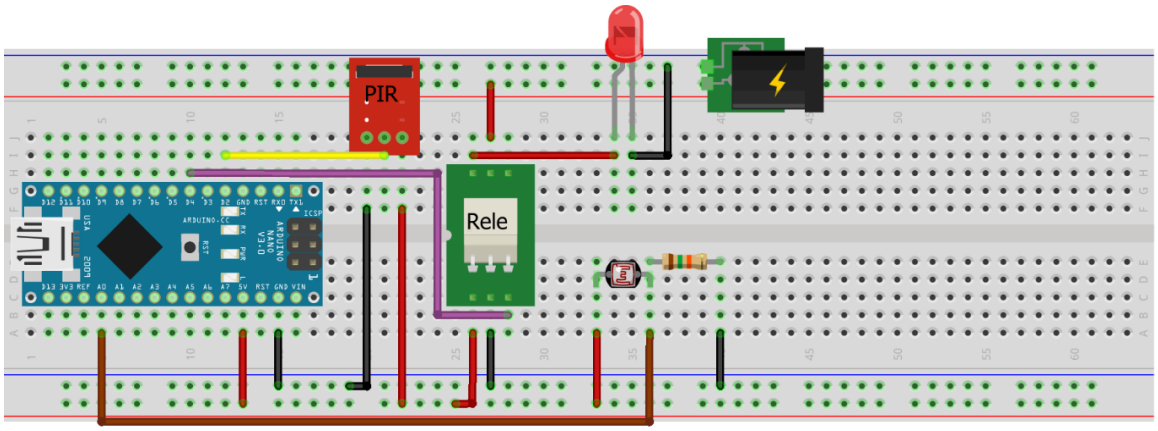
Exemplo de teste do Sensor

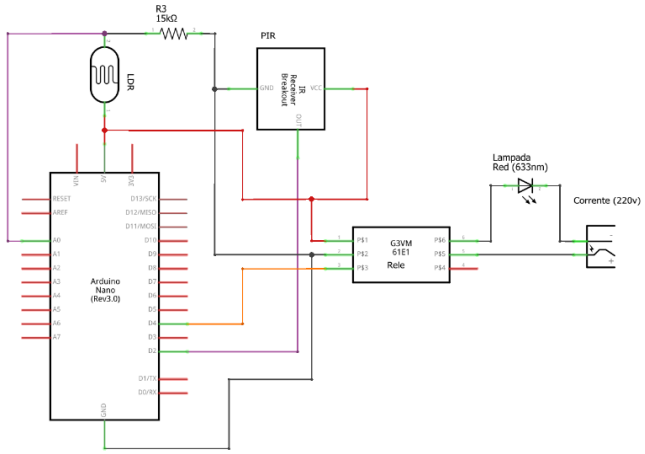
Efetuou-se um teste a este sensor, afim de verificar os seus valores e perceber a sua aplicação. Para isso aproveitou-se o estudo paralelo do PIR, e formou-se uma estrutura de controlo de lâmpadas, automatização.

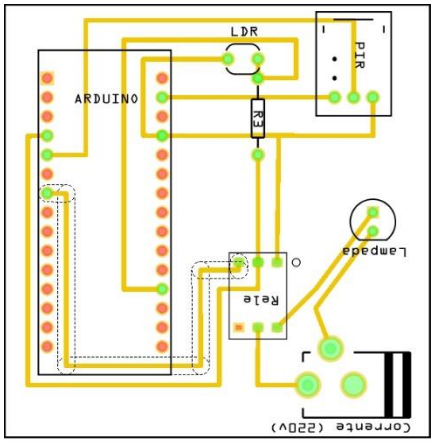
Material usado

1- Arduíno Nano
2- Sensor LDR
3- Sensor PIR
4- Lâmpada – no circuito simulada por Led
5- 1 resistência 15 Kohms
6- 1 Relé
7- Ligação a rede (220V – Atenção a utilização de corrente alternada da rede elétrica)
8- Alguns fios de ligação
9- 1 Breadboard

Circuito utilizado







Código usado

Para tornar o esquema mais apelativo criou-se o seguinte código:

```
/*
Estudo do Sensor de movimento PIR e sensor de luz LDR.
Estes sensores são capazes de fornecer informação de movimentos, no caso do Pir, e intensidade da luz ambiente, no caso do Ldr.
Assim decidiu-se criar uma simulação de controlo de uma lâmpada, mediante algumas condições:
                - Ausência de luz e movimentos
Então mediante estas condições a lâmpada acende, através do acionamento do rele, devido a diferença
de voltagem (lâmpada 220v, Arduíno 5v).
Circuito simples e eficaz para uma poupança de energia.
Data: 05 de outubro de 2019, criado por Nelson com a colaboração de Francisco e Gonçalo.
  */
  // Definir os pinos que serão utilizados no projeto
  #define ldr A0       // pino de leitura digital do sensor luz
  #define pir 2        // pino de leitura digital do sensor movimento
  #define rele 4       // pino de comando do modulo rele
  // Variáveis gerais
  int luz;
  int mov;
  //Inicio do programa
  void setup()
  {
   Serial.begin(9600);
   // Define os pinos dos sensores como saída / entrada
   pinMode(rele, OUTPUT);           // Pino rele
   pinMode(ldr, INPUT);             // Pino de sensor de luz
   pinMode(pir, INPUT);             // Pino sensor de movimento
  }
  void loop()
  {
   // Ler os estados dos sensores e armazenar nas variáveis luz e mov
   luz = analogRead(ldr);
   mov = analogRead(pir);

   if (mov == 1 && luz <= 50)       // Condição: movimentos + pouca luz ---> liga rele (ajuste de sensor de luz)
   {
    digitalWrite(rele, LOW);        // Ligar rele (verificar o nível de acionamento - high ou low)
    Serial.println(mov);            // luz ligada
   }
   else                             // Caso contrário
   {
    digitalWrite(rele, HIGH);       // desliga o rele
   }
  }
```

Explicação do código

1. Na parte inicial definem-se as portas, e as variáveis necessárias para o programa:

```
// Definir os pinos que serão utilizados no projeto
#define ldr A0       // pino de leitura digital do sensor luz
#define pir 2        // pino de leitura digital do sensor movimento
#define rele 4       // pino de comando do modulo rele


// Variáveis gerais
int luz
int mov
```

2. Iniciando o programa, no “setup” inicia-se o serial e define-se as portas identificadas anteriormente como “OUTPUT”:

```
//Inicio do programa
void setup()
{
  Serial.begin(9600);
  // Define os pinos dos sensores como saída / entrada
  pinMode(rele, OUTPUT);           // Pino rele
  pinMode(ldr, INPUT);             // Pino de sensor de luz
  pinMode(pir, INPUT);             // Pino sensor de movimento
}
```

3. Agora no “loop” inicia-se as repetições consecutivas, assim como a atribuição da entrada dos dados dos sensores para as variáveis.
   Aplicamos uma condição, com o “IF”, onde a falta de luz e existindo movimentos se liga o rele mediante a configuração Low ou High.
   No monitor serial regista-se os dados da variável “mov”.

```
void loop()
{
  // Ler os estados dos sensores e armazenar nas variáveis luz e mov
  luz = analogRead(ldr);
  mov = analogRead(pir);

  if (mov == 1 && luz <= 50)         // Condição: movimentos + pouca luz ---> liga rele (ajuste de sensor de luz)
  {
    digitalWrite(rele, LOW);           // Ligar rele (verificar o nível de acionamento - high ou low)
    Serial.println(mov);               // luz ligada
  }

  else                                 // Caso contrário
  {
    digitalWrite(rele, HIGH);        // desliga o rele
  }
}
```

Conclusão

- ✓ Estudo do sensor LDR efetuado com sucesso.
- ✓ Aprendizagem e construção de circuitos elétricos.
- ✓ Verificação e exploração de diferentes funções do código no Arduíno.
    - #define – atribuição de um nome a um valor constante
    - int – variável inteira
    - if () – verifica uma condição e executa os respetivos procedimentos
    - analogRead() – faz a leitura de um especifico pino digital
    - digitalWrite() – escreve um valor alto ou baixo num pino digital
    - Serial.begin() – comunicação entre o Arduíno e o pc
    - Serial.println() – imprime a informação com quebra de linha
    - pinMode() – configuração do pino como input ou output
    - LOW - valor baixo
    - HIGH - valor alto

Objetivo concluído com sucesso.

Obrigado pela Atenção.

Informações e especificações do sensor recolhidas no datasheet do sensor, Wikipédia e pesquisa geral.

Para mais informações pode verificar no datasheet:
https://components101.com/sites/default/files/component_datasheet/LDR%20Datasheet.pdf