

# EA075: Projeto-Exemplo

## Lâmpada controlada pela luminosidade

Instrutores: JOSÉ RAIMUNDO DE OLIVEIRA e WU SHIN-TING

---

**05/06 de agosto de 2019**

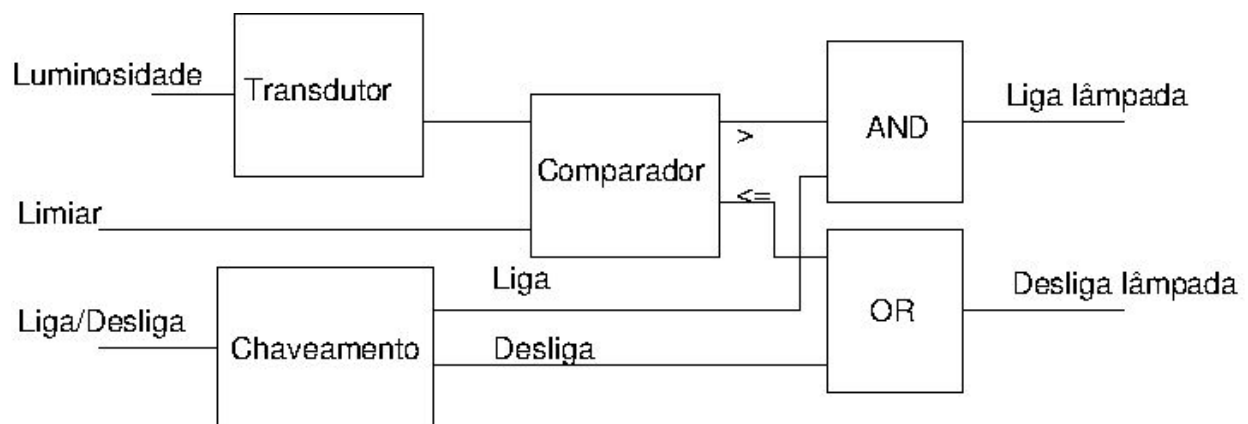
### Especificação Funcional:

**Controle de uma lâmpada** de 12V/7W pela **variação da luminosidade**. Quando a luminosidade estiver abaixo de um valor configurado, a lâmpada acende automaticamente. **O conjunto só opera se ele estiver ligado.**

**12/13 de agosto de 2019**

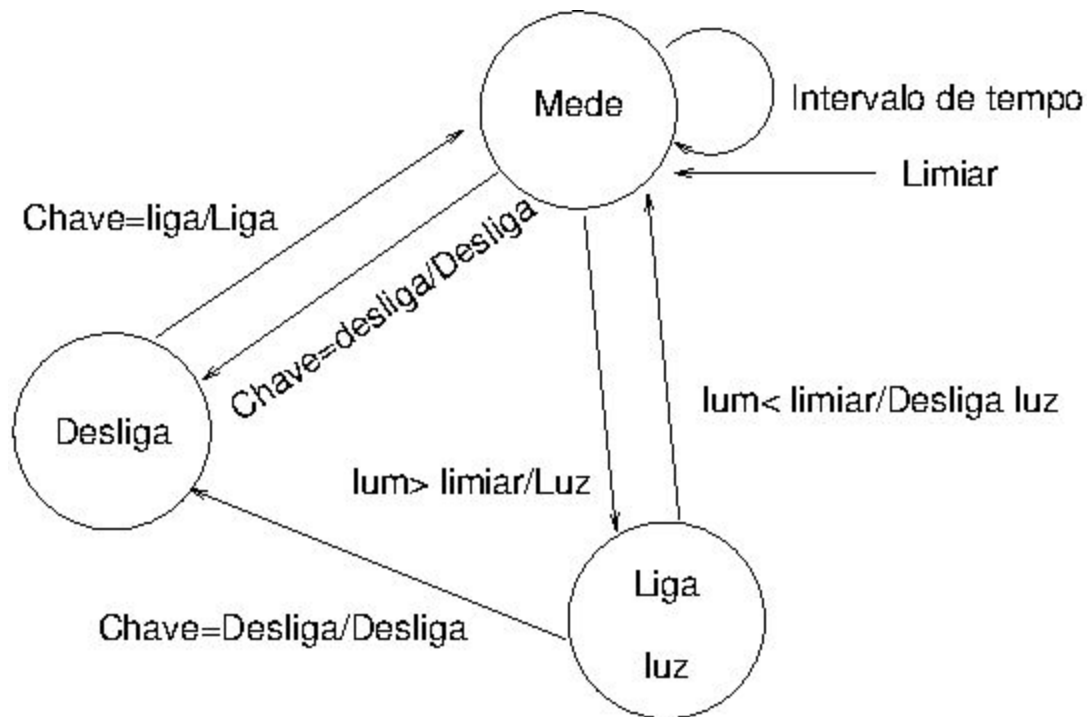
### Modelagem:

Liga/Desliga do usuário é “postergado” depois de algumas condições.



---

**Descrição do comportamento desejado em máquina de estados: (máquina de Mealy)**



**Projeto:**

**Pesquisa:**

- 1) Como operam os sensores de luminosidade?
- 2) Como se converte o sinal de saída do sensor escolhido para sinal elétrico?
- 3) Como traduzir o limiar setado em sinal elétrico?
- 4) Como comparar o sinal do sensor com o limiar?
- 5) Como acionar a lâmpada?
- 6) Há algum projeto similar? Foram encontrados vários projetos. Um baseado somente em componentes eletrônicos é mostrado na Fig. 1, e o outro microcontrolado é apresentado na Fig. 2. Ambos os circuitos são detalhados em [1] e [2], respectivamente. Para quem tem dificuldades em entender as funções da ponte de Wheatstone, do comparador (com realimentação

positiva) A, da chave eletrônica TR1 (circuito de chaveamento) e do dispositivo de proteção D1 ao chaveamento de uma carga indutiva R no esquemático da Fig. 1, consultem [3].

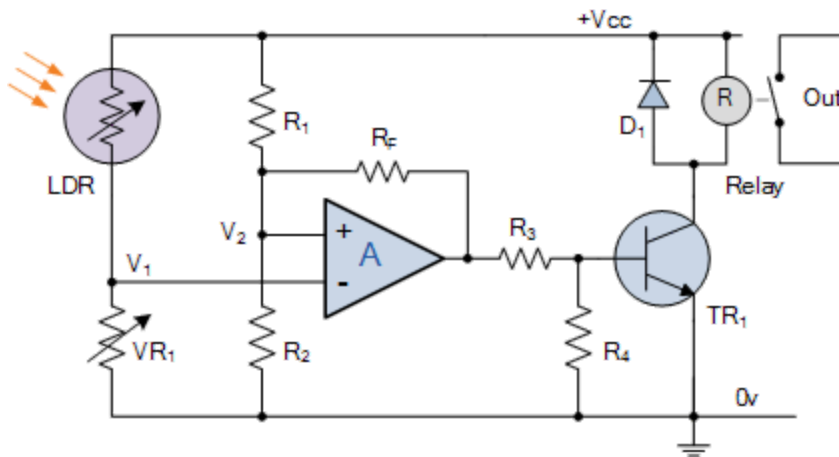


Fig. 1: Implementação com circuitos eletrônicos (Fonte: [1]).

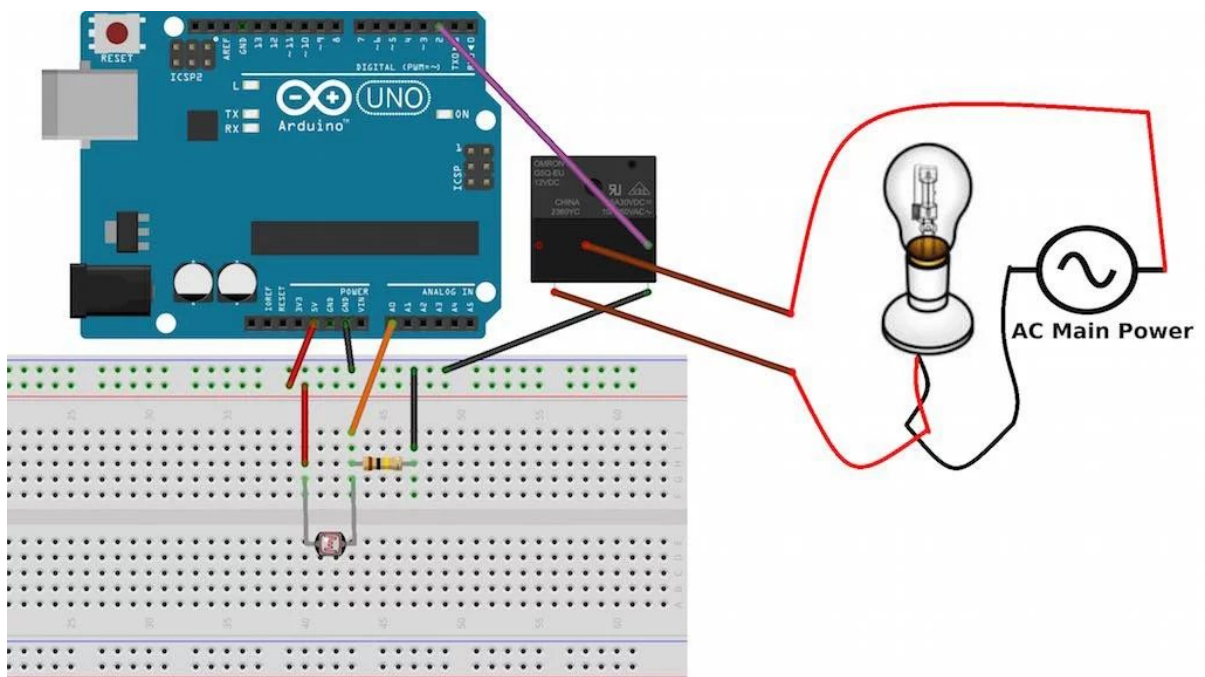



Fig. 2: Implementação com um microcontrolador (Fonte: [2]).

### Restrições do nosso projeto:

- 1) Microcontrolada
  - a) Quais funções serão transferidas para o microcontrolador?
  - b) Qual será a interface entre o microcontrolador e os componentes?
- 2) Componentes disponíveis, além das chaves, resistores, e transistores no almoxarifado da FEEC:

|  |  |  |  |
|--|--|--|--|
|  <p>Potenciômetro 10K</p> |  <p>LDR</p> |  <p>lâmpada automotiva<br/>12V/5W</p> | <p><b>Microcontrolador</b><br/>pinos analógicos (quantidade de canais, p. 464 em [10])<br/>pino GPIO: nível lógico: 0/3.3V até 66 pinos GPIO soma das correntes de todos os pinos &lt; 100mA [5]</p> |
|--|--|--|--|

**15 de agosto de 2019**

**Tomadas de decisão:**

Tomando a Fig. 1 como ponto de partida, é necessário decidir quantas funções podem ser substituídas para termos o comportamento especificado.

- 1) **Qual microcontrolador usar?** [Microcontrolador Kinetis® L Series KL25 MCUs com o núcleo Arm® Cortex®-M0+, montado no *kit* Freedom KL25Z [5]].
- 2) **Como setar o limiar?** Pelo potenciômetro [6] ou por um *trimpot* [7], cuja variação da resistência pode ser convertida para a variação de tensão.
- 3) **Como medir a luminosidade?** Pelo LDR [1], cuja variação da resistência pode ser convertida para a variação de tensão.
- 4) **Como traduzir o sinal luminoso para o sinal elétrico?** O sensor já se encarrega em “traduzir” a intensidade luminosa para resistência. Porém, o

---

pino analógico do microcontrolador requer um sinal analógico [5]. É necessário elaborar um circuito que converte a corrente que passa pela resistência variável em tensão que esteja no intervalo de tensões processáveis pelo microcontrolador. Ao *kit* FRDM KL25Z está integrado uma alimentação analógica de 0 a 3.3V.

- 5) **Como captar a intenção do usuário?** Por uma botoeira em dois estados: 0, desliga; 1, liga. O microcontrolador é provido de pinos de entrada/saída digital (GPIO) [5], de forma que se convertermos os dois estados liga e desliga em dois níveis de tensão lógicos compatíveis com os GPIOs FRDM KL25Z conseguirá processar os dois estados (Tabela 5 em [5]).
- 6) **Como amostrar periodicamente os valores do potenciômetro e do LDR?** Usar um temporizador periódico (*timer*) disponível no microcontrolador.
- 7) **Como comparar o limiar configurado e a intensidade luminosa? Qual é a grandeza física? Qual unidade?** Usar o microcontrolador para fazer as comparações dos dois valores analógicos amostrados, e acionar a lâmpada conforme as condições lógicas estabelecidas. É necessário entender a transdução (intervalo de resistências do LDR [8]), a faixa de resistências do potenciômetro [9], e a função de conversão de tensões analógicas para um código binário (ver Seção 28.6.1.3, p. 503 em [10]).
- 8) **Como acionar a lâmpada?** Por uma chave eletrônico controlável digitalmente ligada em série com a lâmpada. Será que podemos evitar o relé, que apresenta o problema de barulho mecânico de chaveamento e o de gerar um transiente danoso quando se abre o relé (uma carga indutiva)?

### **Circuito de Interface com os pinos do microcontrolador**

- 1) Compatibilidade funcional: alocação dos pinos conforme as funções que eles suportam (ver Seção 10.3.1, p. 161 em [10])
- 2) Compatibilidade elétrica: conexão da botoeira, conexão do sensor, conexão do potenciômetro, conexão da lâmpada.

- 
- 3) Compatibilidade mecânica: conexões entre os componentes. Um circuito totalmente dedicado com os componentes integrados na placa de circuito impresso? Um circuito de interface que suporta variar os sensores resistivos e as cargas ... por meio de *pin headers*?
  - 4) Compatibilidade temporal: tempo de conversão compatível com o tempo de amostragem periódica?

**Sintetizar o projeto deste circuito de interface num esquemático com uso do EAGLE [12,13]. Os estudantes da Unicamp tem acesso livre a uma cópia da versão [11].**

A vantagem de desenharmos o esquemático do nosso circuito no ambiente **Schematic** do aplicativo EAGLE é que, quando finalizamos o nosso esquemático, podemos gerar um arquivo de *layout* da **placa de circuito impresso** (PIC), em inglês *Printed Circuit Board* (PCB), de extensão *.brd*, clicando simplesmente em **“Generate/Switch to board”**. Neste ambiente podemos rotear as conexões e definir a quantidade de camadas, em inglês *layers*. Neste nosso projeto, somente um camada é suficiente.

Note que para que o EAGLE aloque espaços adequados para os componentes do seu circuito, é fundamental que sejam adicionados componentes de encapsulamentos exatamente iguais aos que serão usados na confecção do circuito final durante o desenho do esquemático. No roteamento das trilhas, não se esqueçam de deixar uma margem para a borda. Uma referência mais completa para a versão 5.0 do EAGLE em português [12]/em inglês [13].

### **Fluxo de controle programado**

Como uma parte das funções dos circuitos eletrônicos da Fig. 1, mais especificamente a comparação entre os sinais e acionamento condicional da lâmpada, foi transferida para o microcontrolador, temos que programar estas funções:

- 
- 1) amostragem periódica: fazer as amostragens das tensões do potenciômetro e do sensor, que correspondem às suas resistências ( $V=RI$ ) em cada evento periódico gerado por um temporizador (rotina de serviço).
  - 2) Fluxo Principal:
    - a) Inicialização do conversor A/D, de um *timer* para gerar interrupções periódicas em cada 0.5s.
    - b) Laço infinito:
      - i) Converter os valores amostrados para as resistências físicas (ver Seção 28.6.1.3, p. 503 em[10]);
      - ii) Comparar as resistências,  $R_{pot}$  e  $R_{sensor}$ .
      - iii) Se  $R_{pot} < R_{sensor}$  e estado da botoeira é ligado, então acione a lâmpada.
      - iv) Retorna para (i).

## 29 de agosto de 2019

Os responsáveis do SATE, João Paulo e Bruno, darão uma aula sobre como gerar uma PCI a partir do arquivo gerado pelo EAGLE. Será escolhido um conjunto de arquivos gerber de um dos membros de cada grupo para a fabricação da PCI.

Quem quiser montar o circuito, é só solicitar autorização de acesso ao LE-23 aos professores responsáveis pela disciplina. No LE-23 estão disponíveis todos os equipamentos necessários para o desenvolvimento de um projeto.

---

Para aqueles que nunca tiveram contato com a solda, vale a pena dar uma olhada na página 24 da apostila do Prof. Teodiano [4].

## **Análise:**

As análises acontecem em todas as etapas para evitar surpresas após a montagem do circuito projetado. Numa empresa isso significará desperdício de tempo e dinheiro.

## **Referências**

[1] Electronics Tutorials. Light Sensors.

[https://www.electronics-tutorials.ws/io/io\\_4.html](https://www.electronics-tutorials.ws/io/io_4.html)

[2] Arvind Sanjeev. How to use an LDR Sensor with Arduino.

<https://maker.pro/arduino/tutorial/how-to-use-an-ldr-sensor-with-arduino>

[3] Claudio José Magon. Conceitos Básicos da Eletrônica: teoria e prática.

<http://granada.ifsc.usp.br/labApoio/images/apostilas/apostilaEletronica2018-v1.pdf>

[4] Teodiano Freire Bastos Filho. Apostila de Eletrônica Básica II.

<http://cbeb2020.org/Basica2/apostila.pdf>

[5] Freescale. Kinetis KL25 Sub-Family 48 MHz Cortex-M0+ Based Microcontroller with USB.

<ftp://ftp.dca.fee.unicamp.br/pub/docs/ea076/datasheet/KL25P80M48SF0.pdf>

[6] TDK Corporation. Potentiometer. <https://components101.com/potentiometer>

[7] TDK Corporation. Preset Potentiometer (Trimpot).

<https://components101.com/resistors/preset-potentiometer-trimpot-pinout-datasheet>



---

[8] LDR - folha técnica do fabricante SUNROM Technologies.

<https://pt.scribd.com/doc/128939598/LDR-Datasheet>

[9] Potenciômetro linear até 10k - folha técnica do fabricante.

<https://www.sparkfun.com/datasheets/Components/General/R12-0-.pdf>

[10] Freescale. KL25 Sub-Family Reference Manual.

<ftp://ftp.dca.fee.unicamp.br/pub/docs/ea871/ARM/KL25P80M48SF0RM.pdf>

[11] Tarcísio Silva. Introdução ao Cadsoft Eagle.

<https://www.embarcados.com.br/tutorial-eagle-introducao/>

[12] Lucínio Preza de Araújo. EAGLE versão 5.

<https://www.te1.com.br/2011/06/download-manual-utilizacao-cad-eagle/#>

[13] CadSoft Computer. EAGLE Manual Versão 5.0.

[http://hades.mech.northwestern.edu/images/b/b4/Eagle\\_Manual.pdf](http://hades.mech.northwestern.edu/images/b/b4/Eagle_Manual.pdf)