

# LED BRIGHTNESS CONTROL

**Anderson Weber Schenkel, Grazielle Techio de Mello, Patrícia Gomes Dallepiane, Sandi da Costa Gehm, Taís Rieger Lucchese, Valéria Braidá**

*Universidade Regional do Noroeste do Estado do Rio Grande do Sul - UNIJUÍ*

*Departamento de Ciências Exatas e Engenharias - DCEENG – Ijuí - RS*

anderson.wschenkel@hotmail.com, grazitmello@gmail.com, p.a.tigd@hotmail.com, sandigehm@yahoo.com.br, taisrl2007@hotmail.com, valeria\_braidá@hotmail.com

**Abstract.** Natural resources are increasingly scarce, so the development of sustainable projects should be encouraged in the academic environment in order to train professionals with social and environmental responsibility. In this context, this paper presents a system of LED brightness control. The implemented control strategy uses the PI (proportional and integral), which provides the best use of natural lighting, saving energy and improving the ergonomics in the environment.

**Palavras-chave:** PI controller, brightness, LED.

## 1. INTRODUÇÃO

O presente trabalho visa à exposição dos detalhes de implementação, análise dinâmica e discussão de resultados de um controlador PID para controle de intensidade luminosa, através da modelagem matemática de uma malha de processo a ser controlada. Esse tipo de controlador caracteriza uma técnica de controle onde atuam ações proporcional, integral e derivativa. Em outras palavras, através de uma variável de controle, objetiva-se que o sinal de erro seja minimizado pela ação proporcional, zerado pela ação integral e obtido com uma velocidade antecipativa pela ação derivativa. Porém, esta última não será abordada neste artigo.

## 2. DESCRIÇÃO DO PROJETO

O projeto pode ser dividido em três partes: atuador, sensor e controlador. Serão analisadas a seguir estas três partes em detalhes.

### 2.1 Atuador

O atuador e a planta constituem tão somente num LED (Light Emitting Diode)

de luz visível que terá sua luminosidade controlada pelo circuito.

O processo de emissão de luz do LED é denominado eletroluminescência e é controlado por corrente elétrica. Estas variáveis seguem uma relação quase linear, o que será positivo para o controle deste sistema. Neste ponto, é pertinente lembrarmos que juntamente com o atuador (LED), o sistema aqui desenvolvido será implementado em um ambiente não controlado, ou seja, em uma sala com luminosidade externa ao circuito desenvolvido. Estes parâmetros serão considerados pelo controlador para obter sucesso nos processos de controle.

### 2.2 Escolha do Sensor

O sensor de luminosidade escolhido foi o LDR. Por ser um sensor de baixo custo, está presente em muitos circuitos eletrônicos que necessitam monitorar a luz ambiente. O seu uso é bem simples, onde através de um circuito divisor resistivo a variação de resistência é convertida em tensão e usada pelo circuito de controle.

Para fazer a leitura da variação de luminosidade, ou seja, a variação de resistência do LDR é necessário montar um divisor de tensão.

Inicialmente, para obter a curva característica do LDR, realiza-se a montagem um circuito simples, com um resistor de  $560\Omega$  juntamente com um LED de auto brilho, ligados a uma tensão variável de 0 a 12 V. Através da incidência de luminosidade do LED sobre o LDR, com o auxílio de um ohmímetro foram realizadas as medições da resistência do LDR e, com o auxílio de um luxímetro, foram feitas as

medições da luminosidade para cada nível de tensão, chegando aos seguintes resultados apresentados na Tabela 1.

TENSÃO (V)	LUMINOSIDADE (LUX)	RESISTÊNCIA (Ω)
12	270000	72
11	222000	75
10	200000	79,4
9	165000	84,5
8	143500	92
7	121500	101
6	106000	113
5	73000	134,2
4	42000	174
3	14000	321
2	410	2120
1	410	2120
0	410	2120

Tabela 1: Parâmetros obtidos através do ensaio.  
Fonte: Própria do Autor, 2015.

As medições foram realizadas em ambiente não controlado (laboratório da universidade), justamente onde o circuito final será implementado. Dessa forma, através dos dados obtidos e com a utilização do *software* Excel, obteve-se a curva característica do LDR, conforme apresentada na Figura 1.

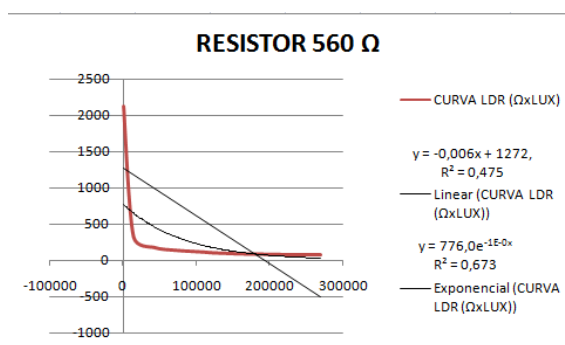


Figura 1: Curva Característica do LDR.  
Fonte: Própria do Autor, 2015.

Através de ferramentas disponíveis no *software* Excel, obtivemos uma função exponencial e uma reta que se aproxima da curva obtida através dos experimentos descritos acima. Sendo assim, a expressão que mais se aproxima da curva do LDR é

uma exponencial, conforme visto na Equação 1:

$$f_t = 776,0e^{-1E0x} \quad (1)$$

### 2.3 Controlador PI

Antes de iniciar o projeto do controlador PI, será realizada uma breve descrição dos elementos principais de um controlador PID. Este controlador possui a combinação das três ações de controle, que são respectivamente: proporcional, integral e derivativa [1].

O controle proporcional atua na resposta transitória do sistema de forma a diminuir o tempo de subida ( $t_r$ ), diminuindo adicionalmente o erro de regime permanente.

Já o controlador integral elimina por completo o erro em regime permanente, no entanto, pode piorar a resposta transitória do sistema.

A ação derivativa apesar de apresentar problemas com ruídos tem o efeito de aumentar a estabilidade do sistema, reduzindo o sobressinal e o tempo de estabilidade, melhorando a resposta transitória.

### 3. PROJETO DO CONTROLADOR

Para a realização do cálculo do controlador, utiliza-se a sintonia por alocação de pólos. A ideia da sintonia por alocação de pólos é modelar um controlador que possua uma resposta em malha fechada que atenda aos requisitos da planta. Desta forma, através de simplificações no modelo devem-se posicionar os pólos do sistema em locais dos planos que atendam aos critérios de sobressinal, tempo de acomodação, dentre outros [2].

A Figura 2 apresenta o diagrama do controlador em malha fechada com a função de transferência da incidência da luminosidade do LED sobre o LDR, obtida através da função exponencial.

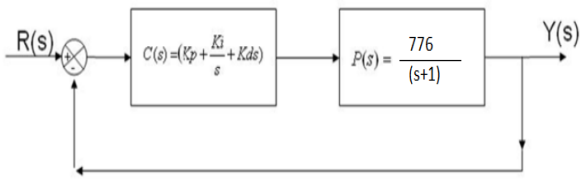


Figura 2: Diagrama em malha fechada.  
Fonte: Própria do Autor, 2015.

Obtendo a função de transferência em malha fechada de acordo com a Equação 2:

$$\frac{y(s)}{R(s)} = G(s) = \frac{P(s)C(s)}{1 + P(s)C(s)} \quad (2)$$

A equação característica é apresentada na Equação 3:

$$\Delta(s) = (1 + 776Kd)s^2 + (1 + 776Kp)s + 776Ki \quad (3)$$

Analisando este sistema através de um de 2ª (pólos dominantes), Equação 4:

$$s^2 + 2\zeta\omega_0 s + \omega_0^2 \quad (4)$$

Resolvendo a igualdade encontram-se os ganhos do controlador:

$$Ki = 0,27 \quad Kp = 1,93 \cdot 10^{-2} \quad Kd = 0$$

#### 4. CONTROLE DIGITAL

O controle digital desse projeto será feito através de um microcontrolador Arduino UNO. Este é uma plataforma de prototipagem eletrônica utilizada para inúmeras aplicações. Neste caso será aplicado para desenvolver, virtualmente, um controlador PI. Como essa é uma plataforma amplamente utilizada no mundo, existem inúmeras bibliotecas disponíveis, entre elas a biblioteca PID\_V1[3].

Basicamente, o controle através do Arduino é feito de acordo com a Equação 5:

$$Saída = Kp e(t) + Ki \int e(t) dt + Kd \frac{d}{dt} e(t) \quad (5)$$

Onde  $e = Setpoint - Entrada$ .

A biblioteca PID\_V1 do Arduino opera da seguinte forma. Inicialmente são definidas as variáveis que serão utilizadas no controle. Após, inicia-se a parte principal do algoritmo. No primeiro momento é definido

um tempo de resposta do sistema, ou seja, um *delay* para ser definido de quanto em quanto tempo o sistema deve recalculer os dados. A partir daí, são computadas todas as variáveis de erro do sistema e é verificada a mudança de tempo para que o sistema recalcule as variáveis. Após, é feito o cálculo do controle PID, seguido da nova atribuição de variáveis que serão utilizadas no próximo *loop* do algoritmo dessa biblioteca.

Além da biblioteca PID\_V1, foi utilizada outra biblioteca chamada PID\_FrontEnd. Essa última faz parte de um método gráfico do *software* Processing, que foi empregado para fazer o processamento PID através do Arduino. Por meio dessa ferramenta computacional pode-se analisar a resposta do sistema projetado através de gráficos. No caso do presente trabalho, é mostrado o gráfico da atuação do LED sobre o LDR em forma de uma curva, onde é definido o *setpoint* e através disso o sistema atua de modo que controle o sinal de saída para que ele fique muito próximo do *setpoint*. A proximidade e a calibração da curva de saída com relação ao *setpoint* são definidas através das variáveis Kp e Ki calculadas através da discretização da função de transferência da incidência de luminosidade do LED sobre o LDR encontrada anteriormente.

Para a parte de análise gráfica, foi utilizada a biblioteca ControlP5, desenvolvida para ser voltada diretamente ao controle digital de inúmeros processos. Através dessa biblioteca é possível fazer alteração de parâmetros para obter melhores ajustes da curva de resposta do controlador PI desenvolvido. Além disso, é possível visualizar graficamente a resposta final do sistema [4].

#### 5. RESULTADOS

Nesse ponto é importante ressaltar que o *setpoint* é a variável que determina o nível de luminosidade que se deseja no ambiente escolhido. Ele é ajustado manualmente pelo usuário do circuito, onde

é calibrado até um nível de luminosidade do LED desejado.

Após testes e implementação final do circuito, foi acoplado um potenciômetro a fim de controlar o *setpoint* manualmente. Dessa forma, não são necessários os ajustes dos parâmetros através do computador, podendo ser realizado manualmente com o auxílio do potenciômetro, até o LED chegar ao nível ao qual o operador do circuito deseja.

Após a montagem em protoboard, foram realizados testes para a verificação da confiabilidade do sistema.

Para testar o circuito como fonte de distúrbio, pode-se utilizar uma lanterna para incidir mais luz ou um anteparo para fazer sombra ao LDR.

Ao termos um distúrbio no processo (por exemplo, sombra), o controlador irá aumentar/diminuir a luminosidade do LED para tentar manter a saída sob controle e nível estabelecido pelo *setpoint*. No caso de fazer uma sombra ao LDR, o controlador irá aumentar a intensidade de luz do LED. Estas variações podem ser visualizadas no PID\_FrontEnd, de forma gráfica na Figura 3.

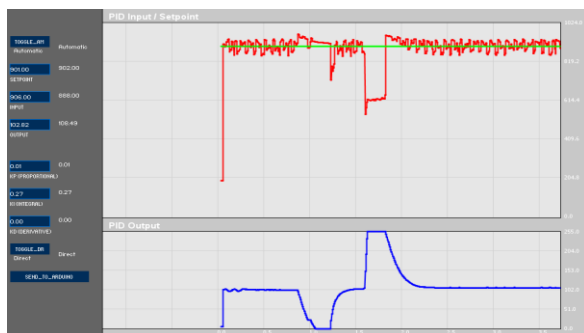


Figura 3: Teste com distúrbios no sistema.  
Fonte: Própria do Autor, 2015.

Na primeira perturbação, foi acrescentado luminosidade sobre o LDR. Nota-se na Figura 3, que a curva vermelha que corresponde a entrada, atingiu um valor maior que o *setpoint*, fazendo com que o Arduino entregasse 0 V ao LED, curva azul. Após, foi retirada a lanterna e o sistema se estabilizou novamente. Logo em seguida, com o auxílio de uma folha de papel, fizemos sombra no LDR, fazendo com que a curva de entrada (vermelha) diminuísse

consideravelmente o valor, e dessa forma, a curva de saída (azul) aumentou até seu limite máximo, entregando 5 V ao LED. Em seguida, com a retirada do distúrbio, o sistema voltou ao normal.

Qualquer alteração feita neste *software* é enviada ao Arduino. Dessa forma, podem-se alterar valores tanto de *setpoint* quanto do controlador PID instantaneamente, dependendo do caso.

## 6. CONCLUSÕES

Neste trabalho, foi desenvolvido um sistema de controle de luminosidade. O sistema respondeu de forma desejada e o controle digital de luminosidade do LED através do sensor LDR foi concluído com sucesso. Para este fim, controlar a luminosidade de um LED, a tarefa está completa.

Porém, para controlar a luminosidade de um ambiente, onde há a necessidade de utilizar lâmpadas, seria indispensável à implementação de um amplificador na saída para acioná-las visto que o Arduino tem limite máximo de saída 5 V.

## 7. REFERÊNCIAS

- [1] Practical Process Control. Disponível em: <<http://www.controlguru.com/pages/table.html>>. Acesso em: 23/11/2015;
- [2] PID Library. Disponível em: <<http://playground.arduino.cc/Code/PIDLibrary>>. Acesso em: 23/11/2015
- [3] MORAES, Marcelo. Controle de Luminosidade com Arduino PID. Disponível <<http://arduinobymyself.blogspot.com.br/2013/04/controle-de-luminosidade-com-arduino-pid.html>>. Acesso em: 23/11/2015
- [4] Control P5. Disponível em: <<http://www.sojamo.de/libraries/controlP5/#installation>>. Acesso em: 23/11/2015