

# Relatório do experimento 4: modulação por largura de pulso

Henrique Koji Miyamoto (RA 169614) e Pedro Luís Azevedo Costa (RA175857)

## 1 INTRODUÇÃO

---

Modulação por largura de pulso (*pulse width modulation* - PWM) é uma técnica para obter resultados analógicos a partir de sinais digitais. Isso é possível ao variar o tempo de alto da onda quadrada de alta frequência gerada pelo sinal digital. Dependendo da largura do pulso, pode-se obter como resultado um valor intermediário entre o nível zero e o nível alto.

O objetivo desse experimento foi montar um modulador por largura de pulso, utilizando o máximo possível de componentes de prateleira. Para tanto, utilizamos os componentes 7493 (contador) e 7485 (comparador magnitude). Nosso modulador por largura de pulso foi projetado no programa Quartus II e testado na placa FPGA, onde a intensidade luminosa de um LED foi controlada de acordo com o número de 4 bits configurado nas chaves.

## 2 MATERIAIS E PROCEDIMENTOS

---

Esse experimento foi projetado e simulado no ambiente virtual do Quartus II e testado na placa FPGA. O projeto consiste em um contador de 4 bits, um comparador de magnitude de 4 bits, um LED, cuja potência será variada, e um conjunto de quatro chaves, que servirá como entrada de dados. A disposição dos componentes é ilustrada no diagrama abaixo.

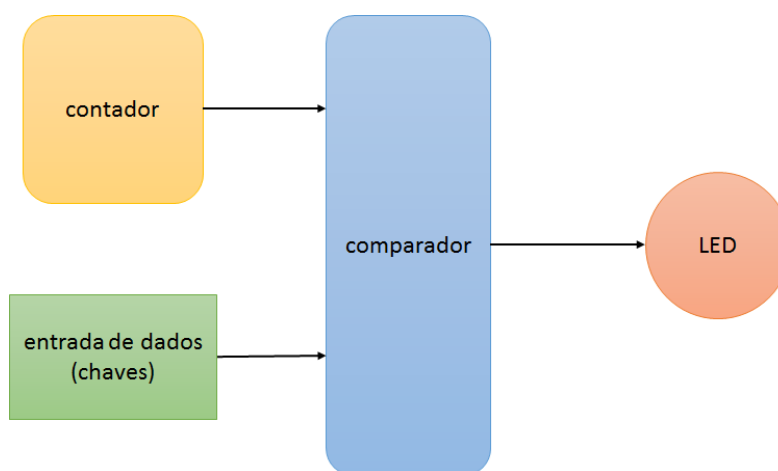


Figura 1. Diagrama de blocos do experimento

A ideia de funcionamento é que o sinal que liga o LED seja ativado apenas para valores do contador menores que o valor selecionado nas chaves, de modo que possamos controlar o *duty-cycle* (razão entre o tempo de alto e o período do sinal gerado). Um fluxograma desse funcionamento está representado a seguir.

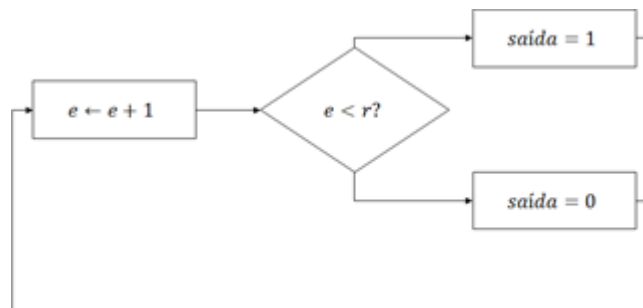


Figura 2. Fluxograma lógico de funcionamento do experimento

Para a montagem do experimento, utilizamos componentes de prateleira (*off-the-shelf*): os CIs 7493 (MOTOROLA, 2015), como contador de 4 bits, e 7485 (NATIONAL SEMICONDUCTOR, 2015), como comparador magnitude de 4 bits.

O contador (CI 7493) funciona da seguinte forma: quando ambas as entradas de controle MR1 e MR2 não estão ativas (isto é, o valor em MR é 0), o dispositivo realiza a contagem de 0 a 15 (4 bits). O resultado da saída é expresso nas portas Q0, Q1, Q2 e Q3, sendo Q0 o bit menos significativo. O dispositivo possui entrada para *clocks* (CP0 e CP1). Internamente, eles são ligados de maneira independente, o que requer uma sutileza na implementação, explicada mais abaixo no texto.

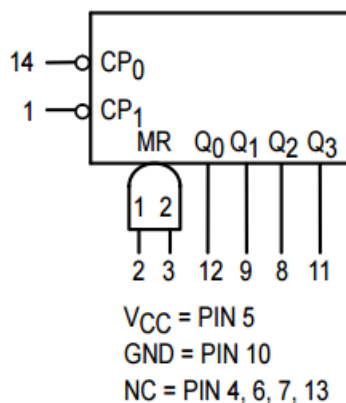


Figura 3. Diagrama do CI 7493 (MOTOROLA, 2015).

O CI comparador de magnitude 7485 recebe como entrada dois vetores de 4 bits (A e B) e tem saídas que indicam se as condições  $A < B$ ,  $A = B$  e  $A > B$  são verdadeiras ou falsas. No caso, só nos interessamos em saber se  $A < B$ . A saída correspondente a esse resultado será ligada ao LED do nosso circuito.

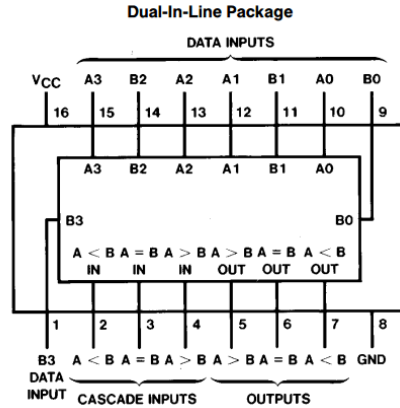


Figura 4. Diagrama do CI 7485 (NATIONAL SEMICONDUTOR, 2015).

O circuito completo é discutido e mostrado a seguir. A entrada *clk* corresponde ao *clock* (disponível na placa FPGA) e as demais 4 entradas, às chaves que selecionam a intensidade do LED, a partir do controle do *duty-cycle*. A configuração das chaves codifica um número binário correspondente ao intervalo de 0 a 15 em decimal.

As duas entradas de controle (RO1 e RO2 no diagrama do Quartus II) do contador não estão ligadas a nada, pois os valores 0 e 0 são suficientes para o dispositivo realizar contagem, conforme explicado acima, na sua especificação. Ainda sobre o contador: devido à sua configuração lógica interna, temos que ligar a saída do bit menos significativo QA à entrada de *clock* CLKB para formar um divisor de frequências, de modo que o sistema funcione como contador de 4 bits.

Nosso circuito tem duas saídas, pois conectamos a saída do comparador a dois LEDs da placa, um vermelho e outro verde, para observarmos melhor o resultado.

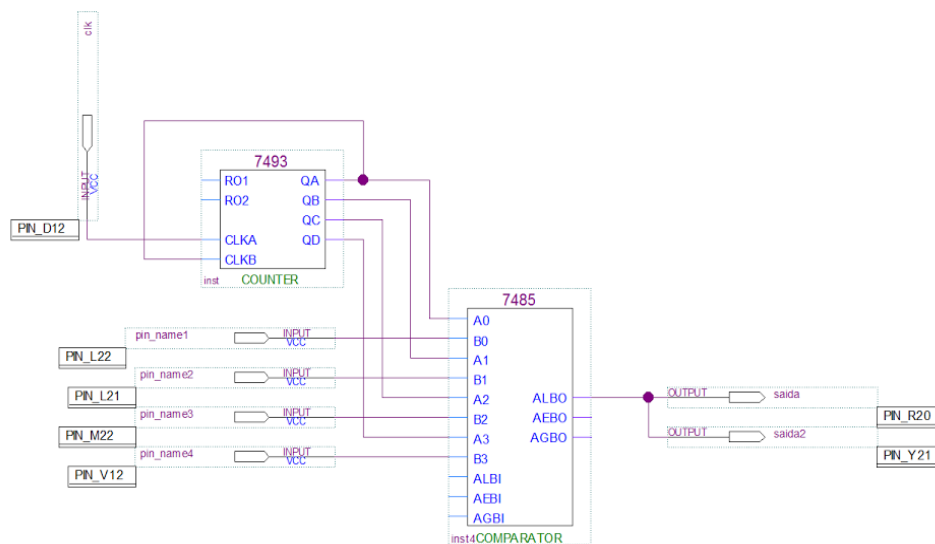


Figura 5. Diagrama do circuito completo, feito no Quartus II

### 3 RESULTADOS

A seguir, apresentamos e discutimos os resultados práticos e das simulações do experimento. O circuito implementado se comportou de acordo com o esperado: o LED apresentou maior luminosidade para entradas com números binários de maior valor. O resultado da simulação funcional é mostrado abaixo. Notamos, como esperado, que, ao aumentar o valor binário da entrada do circuito, o *duty-cycle* aumenta.

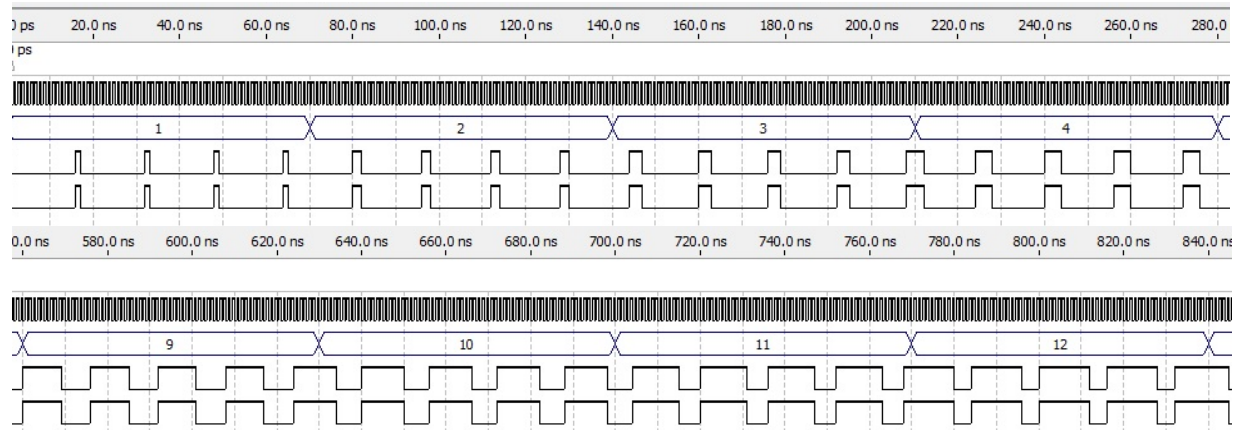


Figura 6. Resultados da simulação funcional

A seguir, apresentamos fotos de alguns testes na placa FPGA para diferentes valores seleccionados nas chaves. As entradas em uso são de SW0 a SW3, em que o bit mais significativo corresponde à entrada SW3 (mais à esquerda). O LED associado à saída do circuito é o mais à direita (LED0).



Figura 7. Teste com entrada de valor 1 (0001)

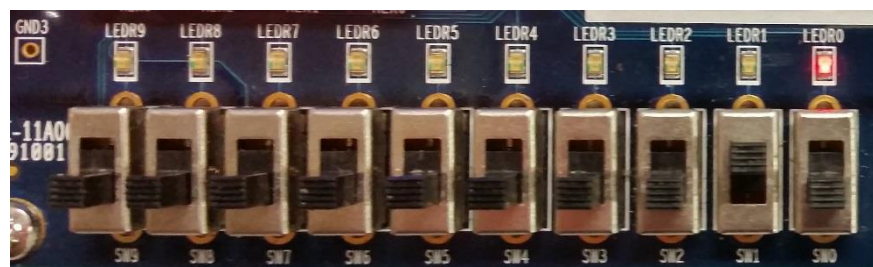


Figura 8. Teste com entrada de valor 2 (0010)



Figura 9. Teste com entrada de valor 5 (0101)



Figura 10. Teste com entrada de valor 15 (1111)

## 4 DISCUSSÃO

O projeto do PWM se mostrou relativamente simples, se comparado aos experimentos anteriores, em parte, porque utilizamos componentes de prateleira, de modo que não tivemos que projetá-los integralmente.

Nas discussões iniciais do projeto, pensamos se deveríamos zerar o contador a cada vez que o valor da entrada fosse igual a zero, ligando a elas uma porta NOR, com saída dividida em dois fios, às entradas RO1 e RO2 do contador. No entanto não o fizemos, pois percebemos que não seria necessário, já que a alta frequência do *clock* torna isso irrelevante para o funcionamento do sistema.

Inicialmente tivemos um problema com o uso da CI com função de contador (7493), pois ele tem entrada para dois *clocks*. Para resolver isso, analisando sua estrutura interna e percebemos que o bit menos significativo recebia a entrada de um *clock* e os bits restantes, de outro. Então ligamos a saída do primeiro bit (Q0) à entrada do *clock* do segundo bit (CLKB), dividindo sua frequência e formando um contador semissíncrono.

Um fator que limita uma melhor análise dos resultados do experimento é que a visão humana percebe a intensidade de luz segundo uma curva logarítmica, ou seja, ao aumentar a intensidade luminosa, torna-se mais difícil distinguir variações de luminosidade. Por isso, a partir de certo valor selecionado nas chaves, fica difícil perceber a diferença de brilho no LED (STOLFI, 2015).

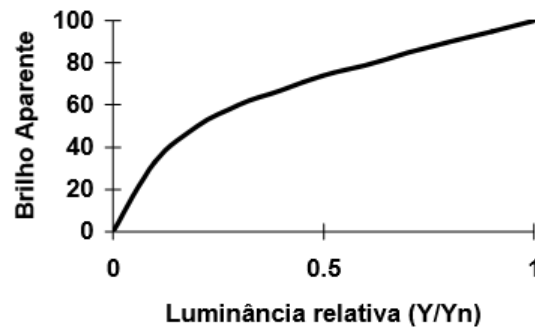


Figura 11. Gráfico de brilho aparente em função de luminância relativa (STOLFI, 2015).

## 5 QUESTÃO EXTRA

---

A corrente que atravessa a lâmpada incandescente nas condições apresentadas é

$$P = Ui \Rightarrow 110 = 110i \Rightarrow i = 1A.$$

A alimentação que usamos para a placa FPGA é através de cabo USB, cuja corrente é de 500mA (UNIVERSAL SERIAL BUS, 2000).

Isso mostra que o problema de conectar uma lâmpada incandescente à placa FPGA é que a corrente fornecida é insuficiente. Para resolver esse problema, poderíamos conectar a lâmpada a uma alimentação externa adequada e fazer com que a placa FPGA apenas controle se esse sinal estaria ativo ou não, sem fornecê-lo diretamente.

Essa ideia pode ser implementada com o uso de um relé. Seu objetivo é exatamente “utilizar pequena quantidade de energia eletromagnética (...) para mover uma armadura que pode gerar uma quantidade de energia muito maior” (HOW STUFF WORKS, 2015). Isso porque “através de uma corrente de controle aplicada à bobina de um relé, podemos abrir, fechar ou comutar os contatos de uma determinada forma, controlando assim as correntes que circulam por circuitos externos” (INSTITUTO NEWTON C. BRAGA, 2015).

Dessa forma, o sinal de saída do PWM controlaria o acionamento do relé, que seria responsável pelo acionamento da lâmpada, podendo, então, controlar sua intensidade luminosa.

## 6 REFERÊNCIAS

---

ALTERA. *Cyclone II FPGA Starter Development Board - Reference Manual*. 2006. Disponível em: <[https://www.altera.com/en\\_US/pdfs/literature/manual/mnl\\_cii\\_starter\\_board\\_rm.pdf](https://www.altera.com/en_US/pdfs/literature/manual/mnl_cii_starter_board_rm.pdf)>. Acesso em: 10 out. 2015.

HIRZEL, Timothy. *Arduino. PWM*. Disponível em: <<https://www.arduino.cc/en/Tutorial/PWM>>. Acesso em: 10 out. 2015.

HOW STUFF WORKS. *How relay works*. Disponível em: <<http://electronics.howstuffworks.com/relay2.htm>>. Acesso em: 10 out. 2015.

INSTITUTO NEWTON C. BRAGA. *Tudo sobre relés (livro completo)*. Disponível em: <<http://www.newtoncbraga.com.br/index.php/como-funciona/597-como-funcionam-os-reles?showall=1&limitstart=>>. Acesso em: 10 out. 2015.

MOTOROLA. *Decade counter; divide-by-twelve counter; 4-bit binary counter*. Disponível em: <<http://pdf.datasheetcatalog.com/datasheet/motorola/SN54LS92J.pdf>>. Acesso em: 5 out. 2015.

NATIONAL SEMICONDUTOR. *5485/DM5485/DM7485 4-Bit Magnitude Comparators*. Disponível em: <<https://www.engineering.uiowa.edu/sites/default/files/ees/files/NI/pdfs/00/65/DS006530.pdf>>. Acesso em: 5 out. 2015.

STOLFI, Guido. *Percepção visual humana. Percepção de luminosidade*. Disponível em: <[http://www.lcs.poli.usp.br/~gstolfi/mack/Ap2\\_PercepVisual\\_M8.pdf](http://www.lcs.poli.usp.br/~gstolfi/mack/Ap2_PercepVisual_M8.pdf)>. Acesso em: 5 out. 2015.

UNIVERSAL SERIAL BUS. *Universal Serial Bus Specification Revision 2.0*. Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips. 2000. Disponível em: <[http://www.usb.org/developers/docs/usb\\_20.zip](http://www.usb.org/developers/docs/usb_20.zip)>. Acesso em: 10 out. 2015.