

**UNIVERSIDADE FEDERAL DA PARAÍBA**  
**CENTRO DE INFORMÁTICA**

GABRIEL ALCÂNTARA, 20160110279

JORDAN ELIAS, 11516379

HELTER YORDAN, 11406573

RELATÓRIO CIENTÍFICO PARA A DISCIPLINA DE CIRCUITOS LÓGICOS II

**CONSTRUÇÃO DE UM SISTEMA DE  
CONTROLE ELETRÔNICO PARA PORTÃO DE  
GARAGEM**

**JOÃO PESSOA, 2018.**

**GABRIEL ALCÂNTARA ROCHA DA FONSECA**

**JORDAN ELIAS RODRIGUES**

**HELTER YORDAN ALVES DA COSTA**

**CONSTRUÇÃO DE UM SISTEMA DE CONTROLE ELETRÔNICO PARA  
PORTÃO DE GARAGEM**

Trabalho solicitado pelo professor Ewerton Monteiro, da disciplina Circuitos Lógicos II, do curso de Engenharia de Computação.

**JOÃO PESSOA, 2018.**

## SUMÁRIO

|   |           |
|---|-----------|
| <b>1.Introdução .....</b>                       | <b>04</b> |
| <b>2.Construção do sistema eletrônico .....</b> | <b>05</b> |
| 2.1 Objetivos .....                             | 05        |
| 2.2 Código Verilog .....                        | 05        |
| 2.3 Materiais utilizados .....                  | 07        |
| 2.4 Etapas da montagem .....                    | 07        |
| <b>3.Conclusão .....</b>                        | <b>11</b> |
| <b>Referências .....</b>                        | <b>12</b> |

## 1. Introdução

Com o avanço tecnológico e a constante evolução maquinária na sociedade contemporânea, o portão eletrônico se tornou uma medida básica de segurança nos dias atuais. Através de uma instalação simples, fazemos com que um portão de grandes ou pequenas dimensões funcione sem necessitar de qualquer tipo de esforço manual.

O funcionamento desse mecanismo, no entanto, é um processo que pode ser bastante complicado para os usuários, deixando a cargo dos profissionais sua instalação e cuidados básicos de manutenção.

Funcionando através de um acionamento de um botão, controlamos a execução desejada do portão, para satisfazer nossa necessidade momentânea, seja abrindo ou fechando o mesmo. Apesar de existir mais de um tipo de motor elétrico, neste trabalho demonstraremos o passo a passo da construção de um motor elétrico para portão deslizante, utilizando-se da placa FPGA DE0-Nano.

No portão deslizante é necessário equilibrar o peso do mesmo e a força necessária para movimentá-lo. Um portão muito pesado demanda uma capacidade maior do motor para que a execução seja correta e não implique em problemas para o usuário. O seu funcionamento se dá através de um controle que irá acionar o motor. Por sua vez, o motor irá acionar uma engrenagem que está conectada no trilho. Assim, a engrenagem faz com que o portão realize seu movimento retilíneo, seja puxando ou empurrando o mesmo.



Figura 1

## 2. Construção do sistema eletrônico

### 2.1 Objetivos

O conteúdo desse trabalho está focado, principalmente, em desenvolver um controle de portão eletrônico funcional valendo-se das utilidades da placa FPGA (Field Programmable Gate Array) DE0-Nano. Aqui iremos expor o embasamento técnico e teórico necessário para a compreensão das etapas da montagem a seguir, tanto na parte física do portão, quanto na parte de programação da placa.

A escolha desse estudo tem como finalidade demonstrar a capacidade da placa FPGA DE0-Nano para as mais diversas funções. Aqui vemos como ela é capaz de controlar o funcionamento de LEDs, que nesse trabalho utilizamos para demonstrar o andamento funcional do portão, além de controlar os botões, que passam o comando desejado para o motor do portão.

A combinação de programação em Verilog, da placa FPGA DE0-Nano, e dos conhecimentos adquiridos na disciplina de Circuitos Lógicos II, foram os requisitos necessários para a execução do plano de trabalho do fim da disciplina, como será demonstrado no decorrer deste relatório.

### 2.2 Código Verilog

```
1  module projeto(CLOCK_50, KEY0, KEY1, LED, GPIO0, GPIO1);
2
3      input CLOCK_50;
4      input KEY0, KEY1;
5      output reg[7:0] LED = 8'b00000000;
6
7      reg[31:0] contador = 0;
8      reg novo_clock;
9      reg[3:0] led_atual = 0;
10
11      reg botao_KEY0 = 1'b1;
12      reg botao_KEY1 = 1'b1;
13
14      reg i = 1'b0;
15
16      output GPIO0;
17      output GPIO1;
18
19      assign GPIO0 = ~(~botao_KEY0 & ~LED[7]);
20      assign GPIO1 = ~(~botao_KEY1 & LED[0]);
21
22  endmodule
```

Figura 2

Nessa primeira parte, estão sendo criadas as variáveis de entradas e saídas necessárias para o funcionamento do código em um único módulo. Os LEDs foram inicializados desligados (0). O contador e a variável led\_atual também foram inicializadas em 0. O reg novo\_clock foi criado para deixar o acionamento dos LEDs numa velocidade visualmente aceitável. É preciso frisar que, os botões funcionam em borda de descida do clock, assim eles precisam ser inicializados em alto (1), para que comecem desligados. As saídas digitais GPIOs

são necessárias para passar o valor do botão para a Ponte H utilizada no projeto. Também é necessário fazer um aterramento na Ponte H com a placa FPGA assim, fecharemos o circuito quando as GPIO enviarem a tensão à Ponte.

```
23 /----- NOVO CLOCK -----/
24
25
26 always@(posedge CLOCK_50) begin
27     contador = contador + 1;
28     novo_clock=contador[24];
29 end
30 /-----/
31
32
33 /----- CONTROLANDO A BORDA DE DESCIDA DO BOTAO DE ABERTURA -----/
34
35
36
37 always@(negedge KEY0) begin
38     botao_KEY0 = ~botao_KEY0;
39 end
40 /-----/
41
42
```

Figura 3

No primeiro always um novo clock é criado, diminuindo a velocidade do pulso, ou seja, controlando a velocidade do novo\_clock. No segundo, está sendo checada a borda de descida do botão. A cada pressionamento, os LEDs são acionados, invertendo o valor da variável botão.

```
45
46 /----- CONTROLANDO A BORDA DE DESCIDA DO BOTAO DE FECHAMENTO -----/
47
48
49 always@(negedge KEY1) begin
50     botao_KEY1 = ~botao_KEY1;
51 end
52 /-----/
53
54
```

Figura 4

No always dessa imagem, é checada a borda de subida do botão. A cada pressionamento, os LEDs são acionados, invertendo o valor da variável botão.

```

56  /----- CONTROLANDO O ACIONAMENTO DOS LEDS -----/
57
58
59  always@(posedge novo_clock) begin
60
61      if (botao_KEY0 == 0)begin
62          LED[led_atual] = 1'b1;
63
64          if (led_atual<7)begin
65              led_atual = led_atual+1;
66          end
67
68          if (led_atual == 7) begin
69              botao_KEY0 = ~botao_KEY0;
70          end
71      end
72
73
74      if (botao_KEY1 == 0)begin
75          LED[led_atual] = 1'b0;
76
77          if (led_atual>0)begin
78              led_atual = led_atual-1;
79          end
80
81          if (led_atual == 0)begin
82              led_atual = 0;
83          end
84      end
85
86  end
87
88  /-----/
89  end
90
91  endmodule
92

```

**Figura 5**

Na última parte do código, está sendo controlado o acionamento dos LEDs. Os LEDs são acionados para que acendam em sequência, simulando o comportamento de um portão. No primeiro if, os LEDs simulam a abertura do portão. Quando o botão\_KEY0 é pressionado, enquanto o led\_atual for menor que 7, ou seja enquanto o led\_atual não chegar ao último LED, ele irá ser incrementado, ligando todos os LEDs. Já o segundo if trata do fechamento do portão. Quando o botão\_KEY1 é pressionado, o led\_atual será decrementado, ocasionando o apagamento de todos os LEDs.

## 2.3 Materiais utilizados

Para a construção do portão eletrônicos prevista neste trabalho foram necessários alguns materiais específicos. São eles:

- Placa FPGA DE0-Nano;
- Motor DC;
- Ponte H;
- Bateria de 9V

## 2.4 Etapas da montagem

1. Conectar a placa FPGA DE0-NANO na Ponte H, através dos GPIOs designados para receber o valor dos botões utilizados e o aterramento GND. No caso específico desse projeto, IN1 e IN2 foram os pinos utilizadas para controlar o sentido de rotação. Out1 e Out2 são as conexões com o motor e não importa a polaridade. Para o GND foi

conectado o terra do GPIO e também o negativo da bateria, e no 12V foi inserido o positivo da bateria.

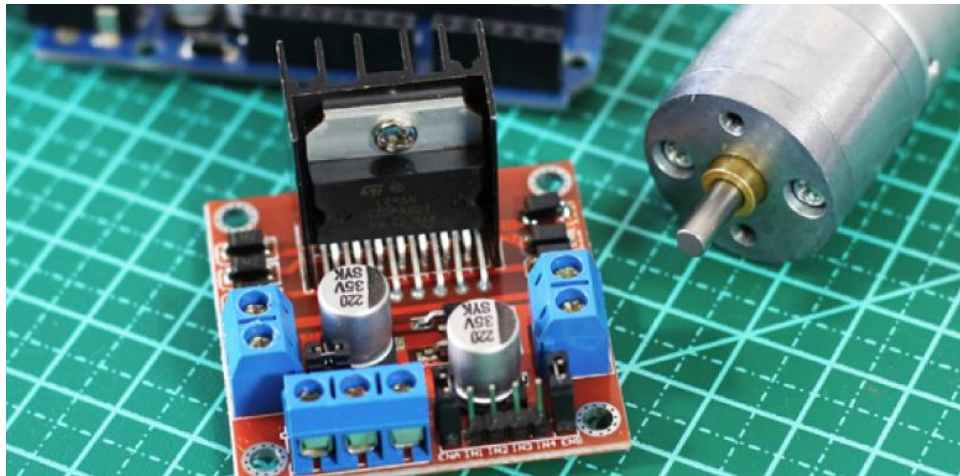


Figura 6

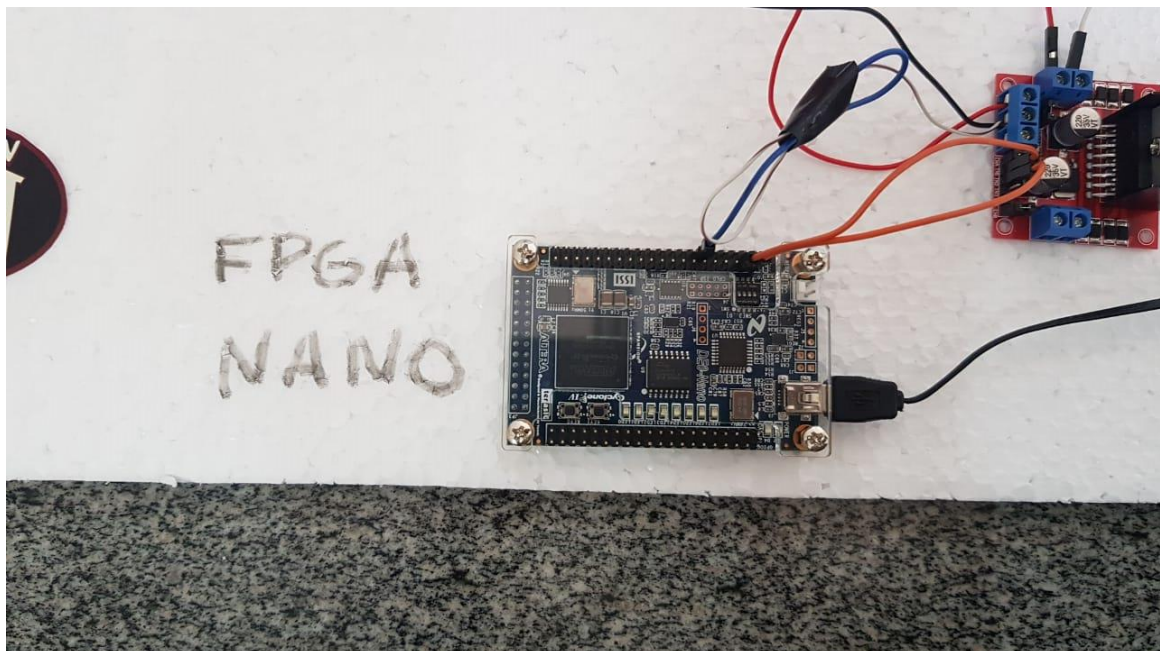
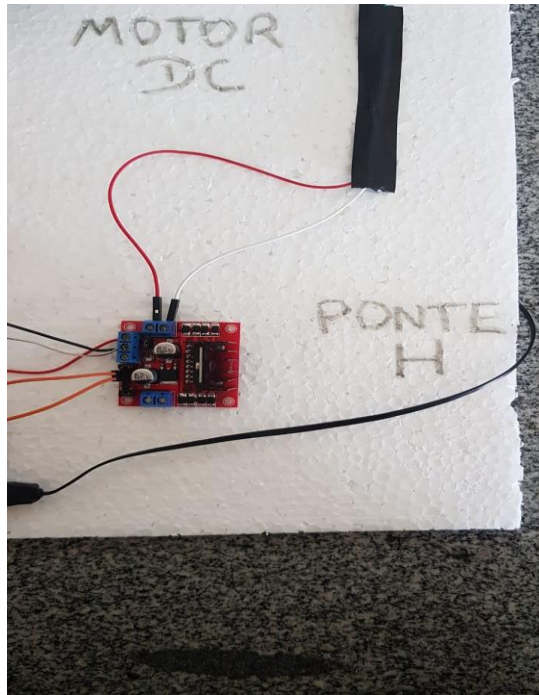


Figura 7

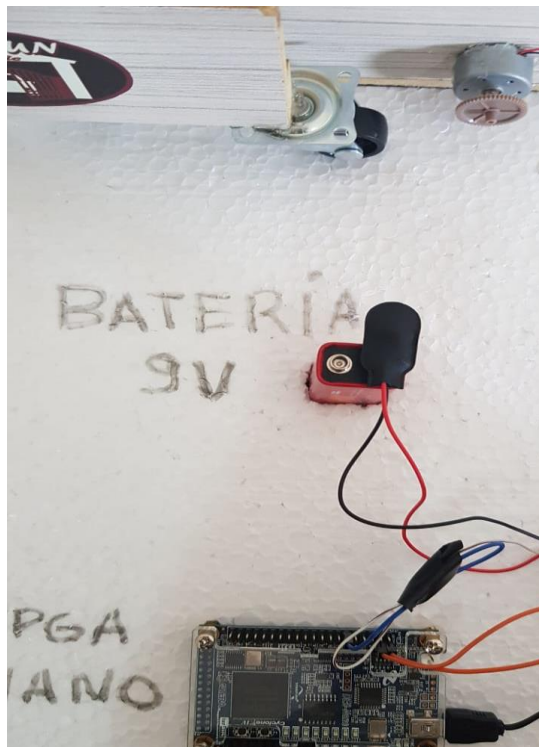
2. Fazer a conexão da Ponte H com o Motor DC (Direct Current), para evitarmos a indesejada tensão inversa de retorno em direção a placa FPGA, evitando danos, e controlarmos o sentido de rotação do motor. A ponte também permite controlarmos a velocidade de giro, porém não utilizamos nessa montagem, então deixamos os pinos responsáveis por essa ação inutilizados.





**Figura 8**

3. Fizemos a alimentação externa com a bateria de 9V na Ponte H, já que o motor funciona com uma tensão mínima (5v) maior do que a fornecida pela placa FPGA (3,3v).



**Figura 9**

4. Projeto montado:

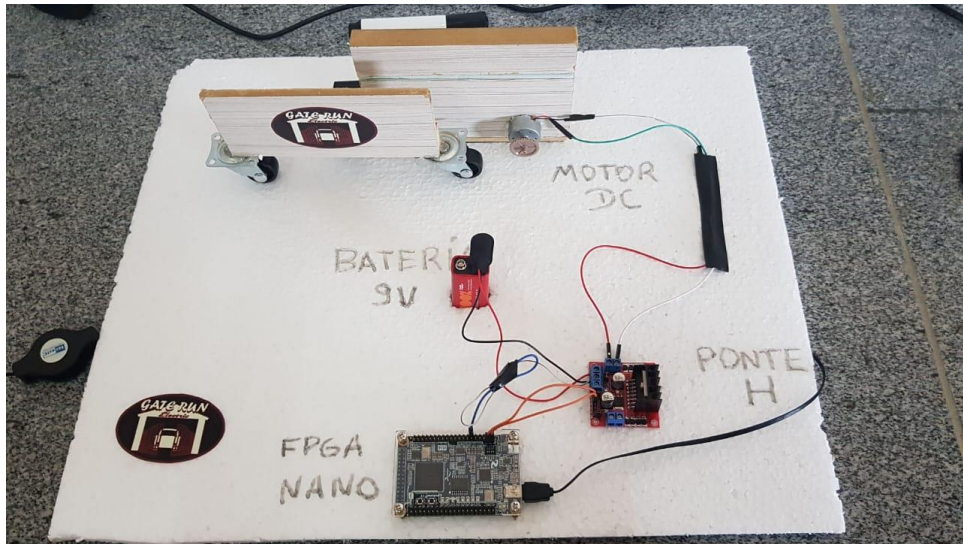


Figura 10

### **3. Conclusão**

Este relatório foi baseado em um experimento onde se buscava retratar o funcionamento de um portão eletrônico deslizante através de controle feito pela placa FPGA DE0-Nano. Para isto, é necessária a existência de um motor que acione uma engrenagem conectada a uma régua, chamada de cremalheira, que nada mais é do que um trilho dentado. Mediante o acionamento deste, essa engrenagem faz com que a cremalheira realize seu movimento retilíneo, empurrando ou puxando o portão.

Logo, é visto que o motor é o componente vital para o funcionamento normal de um portão eletrônico. Apesar dos trilhos ajudarem na abertura e fechamento do portão, a força realizada pelo motor nos priva da necessidade de realizar esse trabalho manualmente. Assim, é necessário que o motor tenha a capacidade de movimentar todo o peso do mesmo.

Durante o experimento, o motor eletrônico para portão foi desenvolvido com o auxílio da placa FPGA DE0-Nano. Através desta, foi possível realizar a programação do circuito, visando atender as demandas referentes a todo comportamento do portão. Entretanto, a placa individualmente não é capaz de fornecer a energia necessária para a alimentação do motor DC. Para auxiliar a placa FPGA, foram utilizados na simulação componentes como a bateria de 9V, para fornecer uma tensão maior do que a original da placa, fazendo possível o funcionamento do motor e a programação em Verilog permitiu adicionar funcionalidades que venham de encontro as necessidades dos usuários do portão como o controle de movimentos (Avançar, retroceder e espera) e o sistema de verificação implementado através do uso de LEDs. Com isso, o funcionamento do motor de portão eletrônico deslizante tornou-se possível e a simulação pôde ser concluída com êxito.

## REFERÊNCIAS

BLOG NETALARMES. **Como funciona o motor elétrico para portão.** Disponível em: <<http://blog.netalarmes.com.br/como-funciona-motor-eletrico-portao/>> Acesso em 30 de Outubro de 2018

BRINCANDO COM IDEIAS. **Módulos para Arduino - Vídeo 13 – Ponte H L298.** Disponível em: <<https://www.youtube.com/watch?v=D13JOucGajI&feature=youtu.be>> Acesso em 20 de Outubro de 2018

FONSECA, Rubens. **Motor para Portão Eletrônico: O Guia Completo para Iniciantes.** Disponível em: <<https://blogdocftv.com/motor-para-portao/>> Acesso em 31 de Outubro de 2018

THOMSEN, Adilson. **Motor DC com Driver Ponte H L298N.** Disponível em: <<https://www.filipeflop.com/blog/motor-dc-arduino-ponte-h-l298n/>> Acesso em 22 de Outubro em 2018