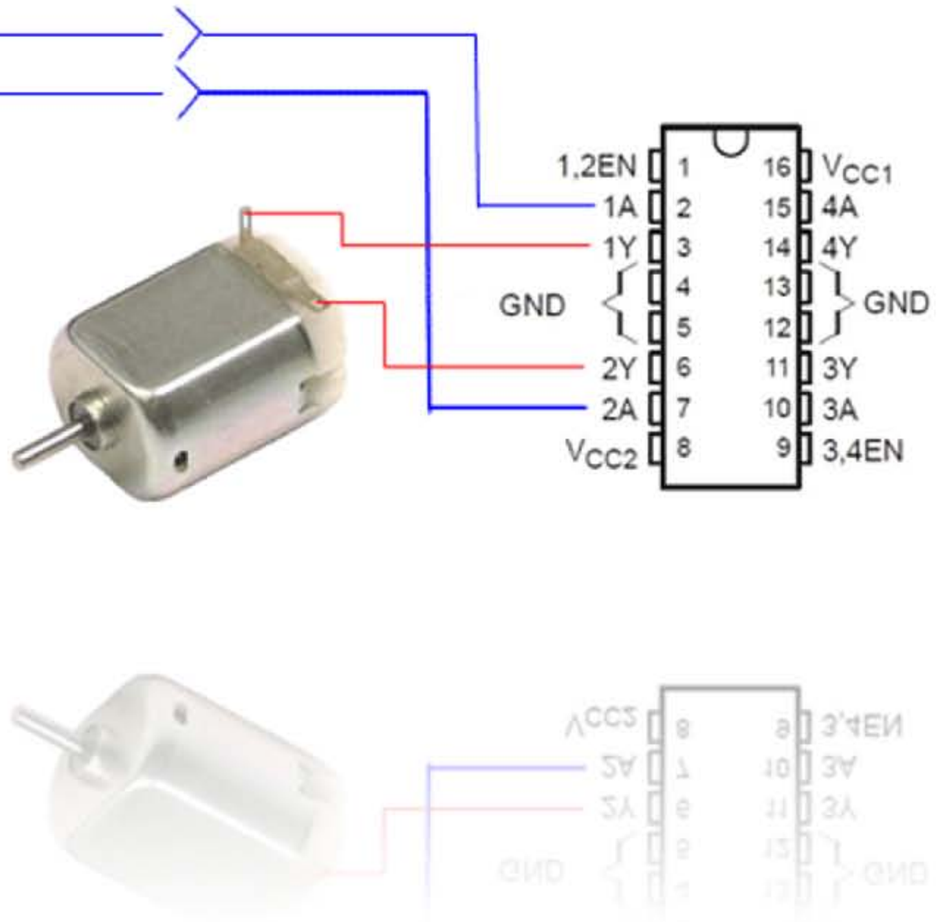
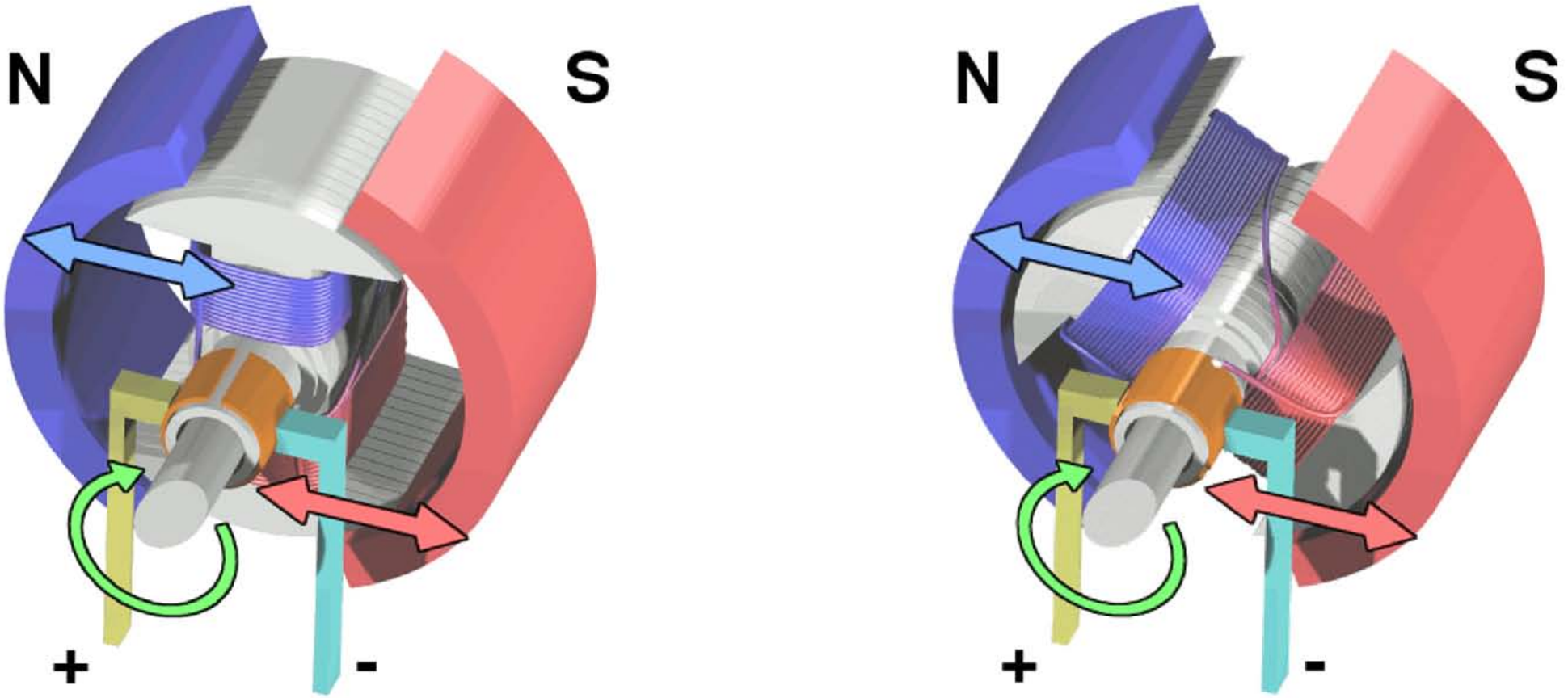
PWM ARDUINO

Pulse Width Modulation

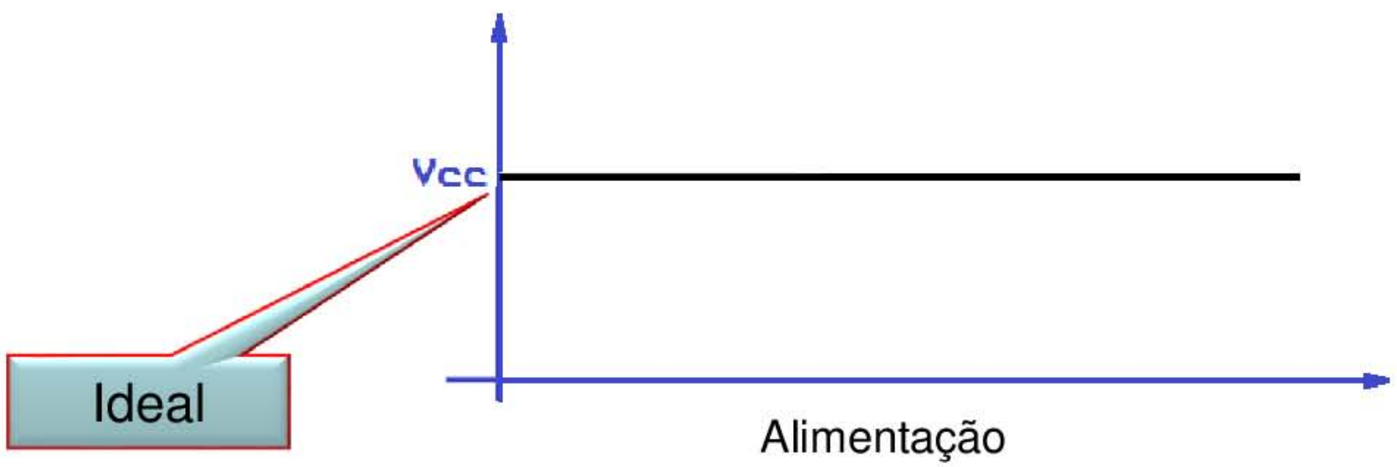


© Prof. Engº esp Luiz Antonio Vargas Pinto
www.vargasp.com
Revisado Novembro 2020

Características

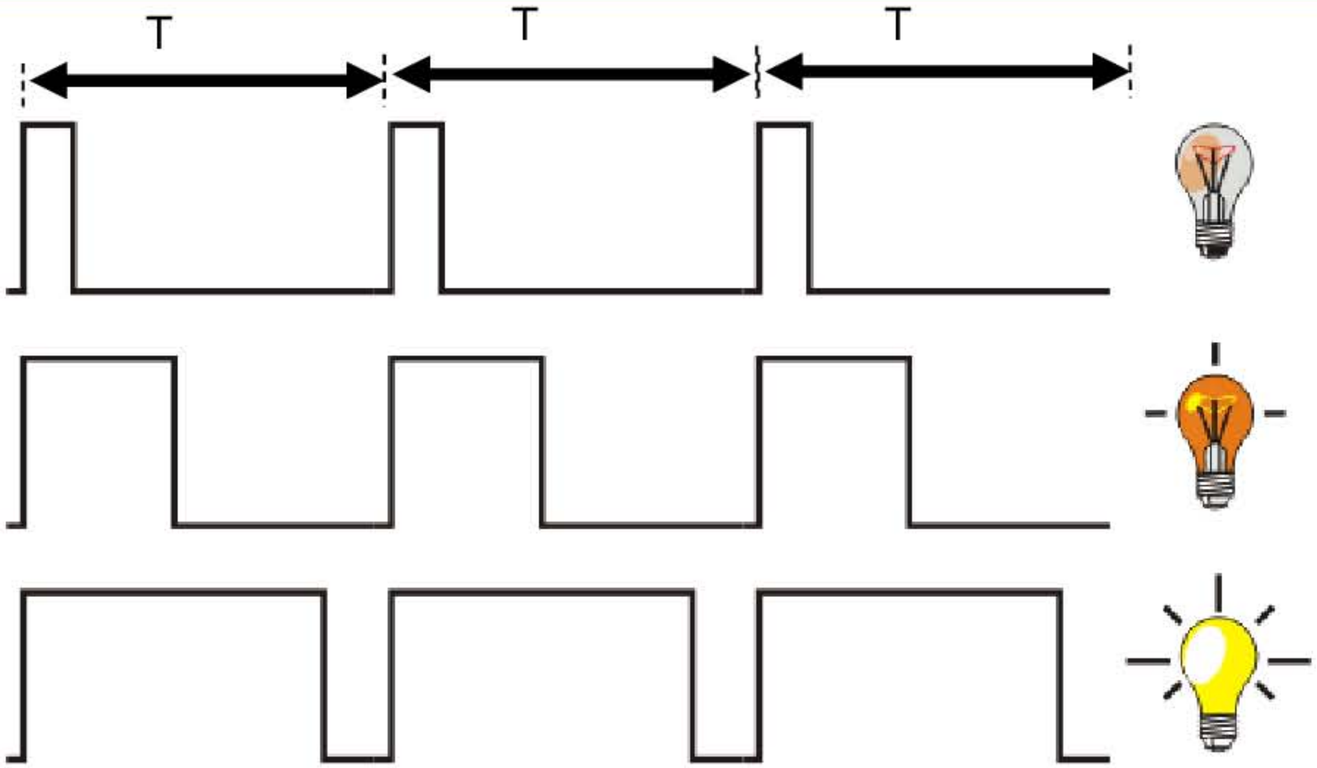


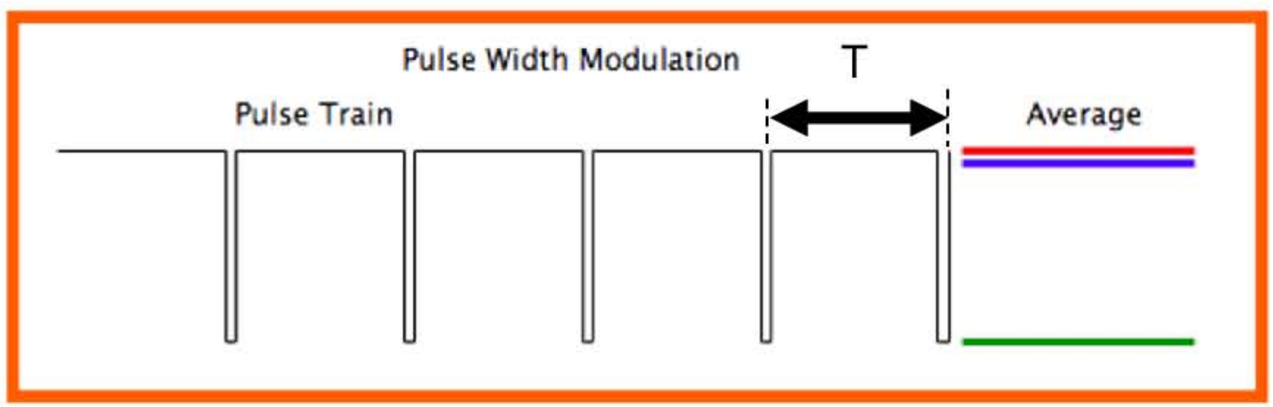
Rotação em RPM



© Prof. Vargas

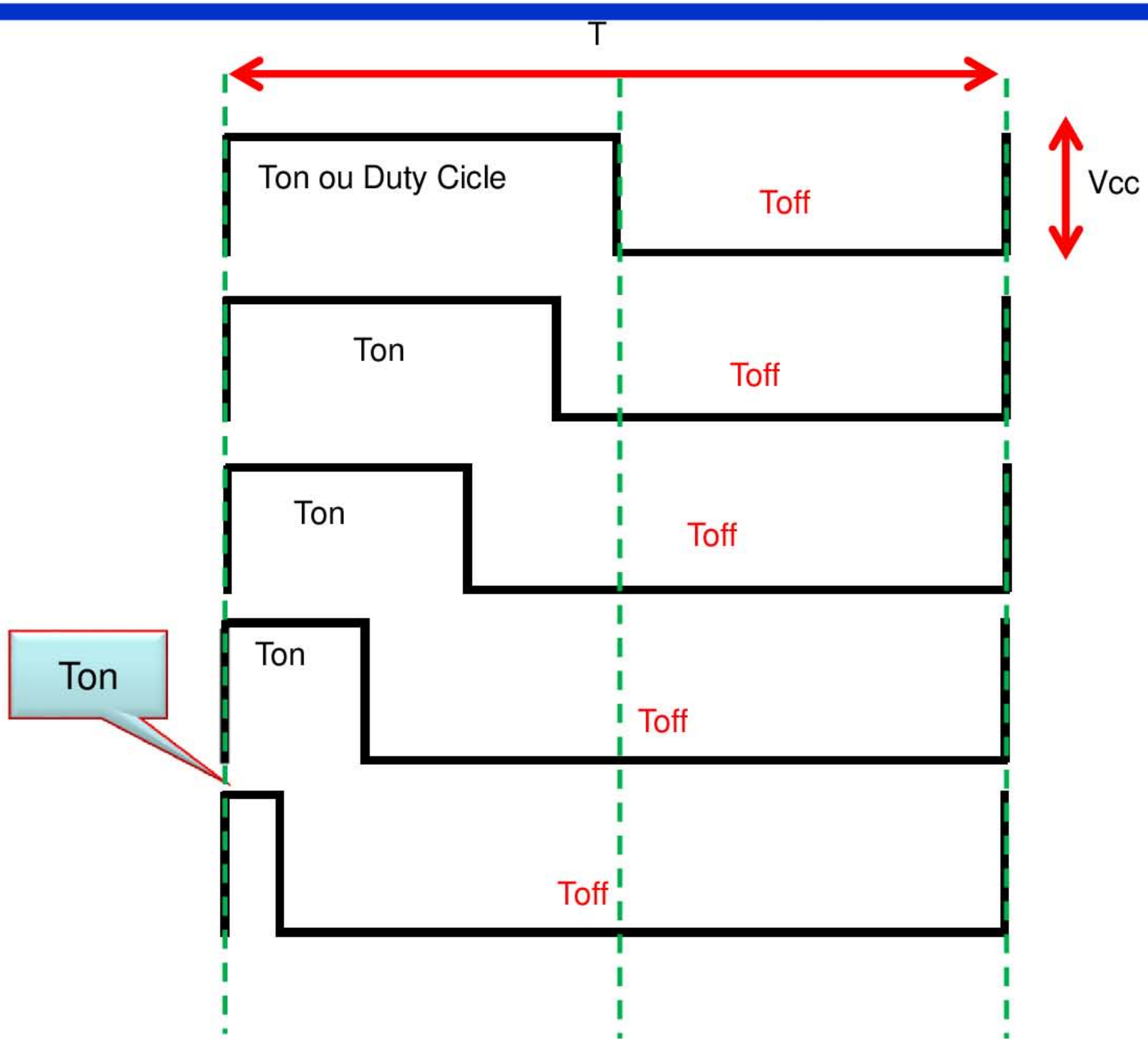
Processo





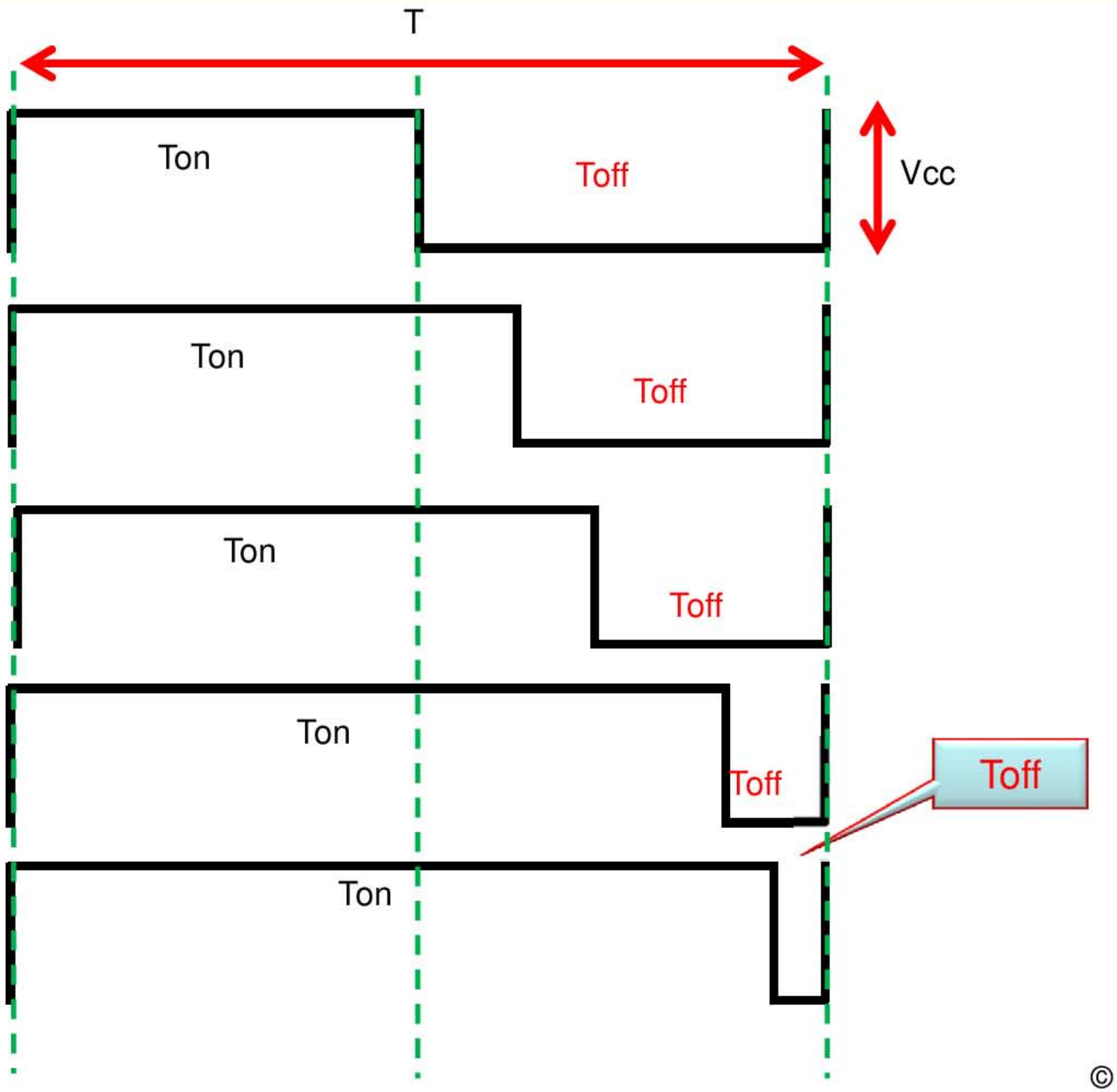
© Prof. Vargas

Intervalando...



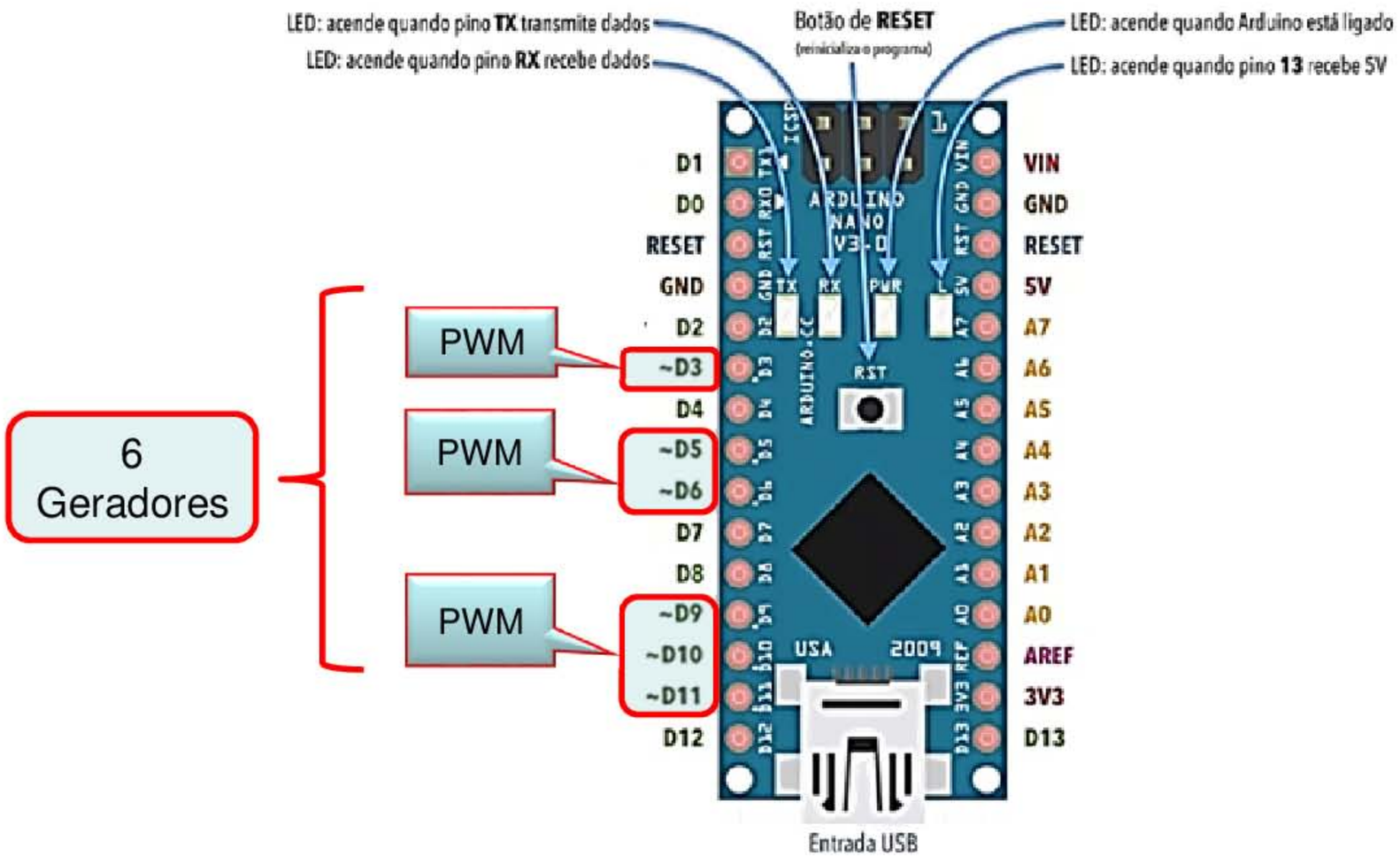
© Prof. Vargas

ou...



© Prof. Vargas

Características

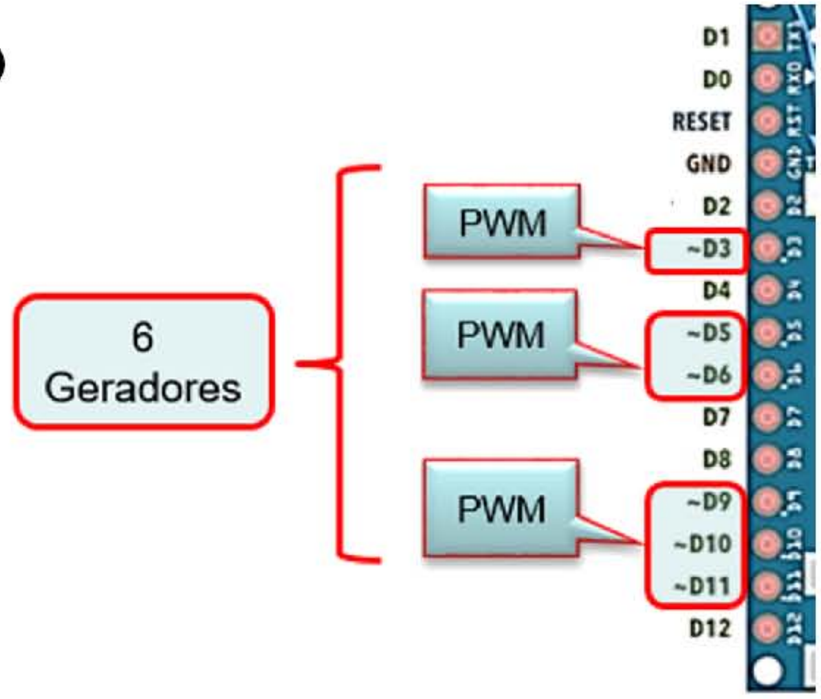


http://eletronicaparaartistas.com.br/arduino-2-configuracao-do-arduino-nano/

© Prof. Vargas

Usando o Nano

- Como se pode observar, o Arduino nano usa os pinos 3,5,6,9,10 e 11 para saída PWM.
- A plataforma tem a função **analogWrite()** para usar PWM.



© Prof. Vargas

analogWrite()

- A função analogWrite() escreve o valor do PWM em um daqueles pinos.
- Após a chamada da função analogWrite(), no pino surge uma onda quadrada de frequência fixa e com duty cycle de 0 a 255 informado na função.
- A frequência dessa onda, na maioria dos pinos é em torno de 490 Hz, porém, os pinos 5 e 6 operam em 980 Hz.

© Prof. Vargas

Sintaxe

analogWrite(pino, valor);

- ☒ pino = pino que gera o sinal PWM - 3,5,6,9,10 e 11
- ☒ valor = duty cycle - ciclo útil do sinal - valor de `0` ↔ 255

© Prof. Vargas

Considerações

- Na maioria das placas Arduino com ATmega168 ou ATmega328 essa função está nos pinos 3, 5, 6, 9, 10 e 11.
- No Arduino Mega, funciona nos pinos 2-13 e 44-46.
- Arduinos mais antigos as placas com um ATmega8 suportam apenas analogWrite() nos pinos 9, 10 e 11.

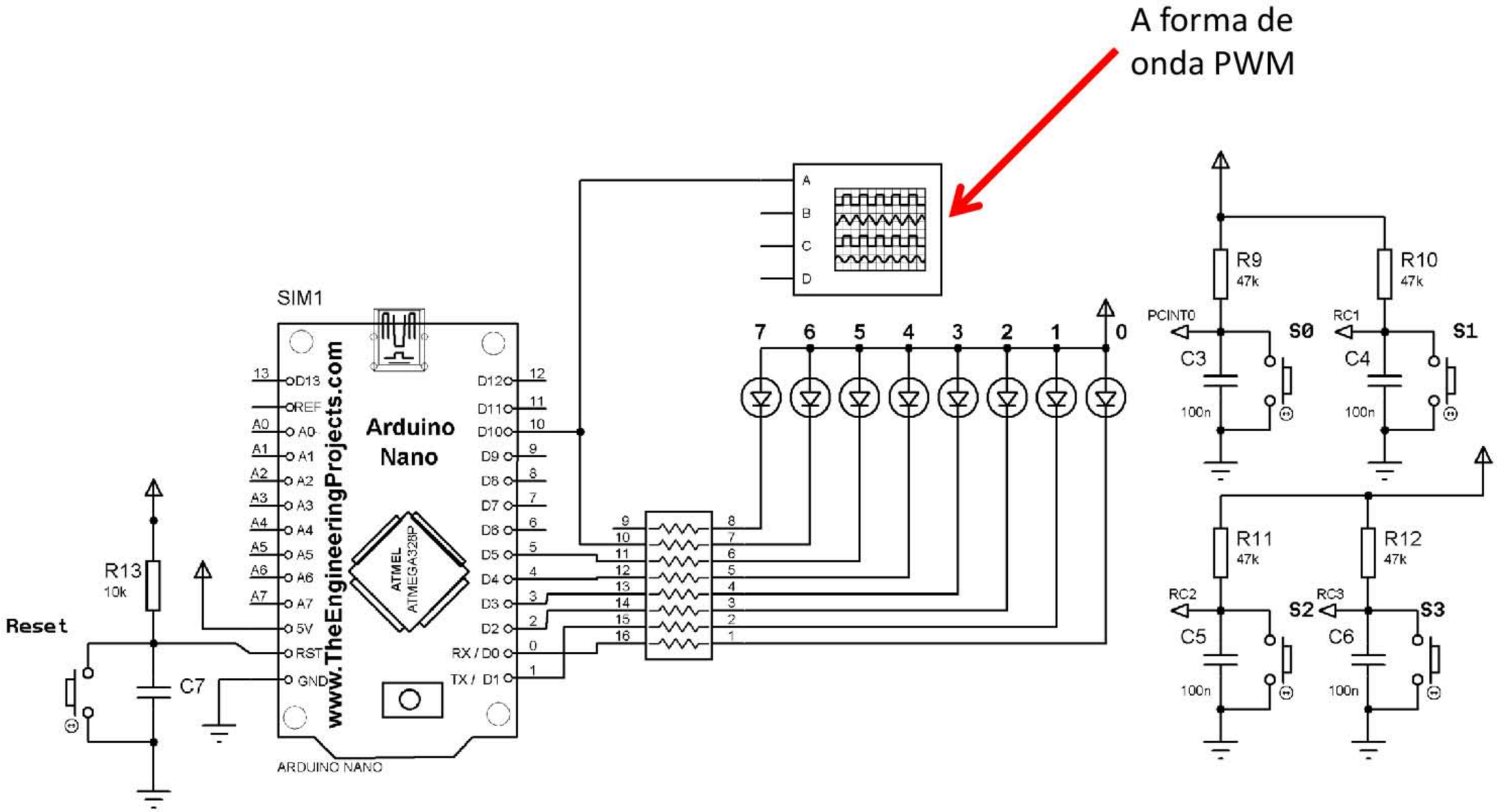
© Prof. Vargas

Notas e problemas conhecidos

- As saídas PWM geradas nos pinos 5 e 6 terão Duty Cycle acima do esperado.
- Isto ocorre devido às interações com as funções millis () e delay (), que compartilham o mesmo timer interno usado para gerar essas saídas PWM.
- Isso será notado principalmente nas configurações de Duty Cycle baixo (por exemplo, 0 - 10) e pode resultar em um valor de 0 não desligando totalmente a saída nos pinos 5 e 6.

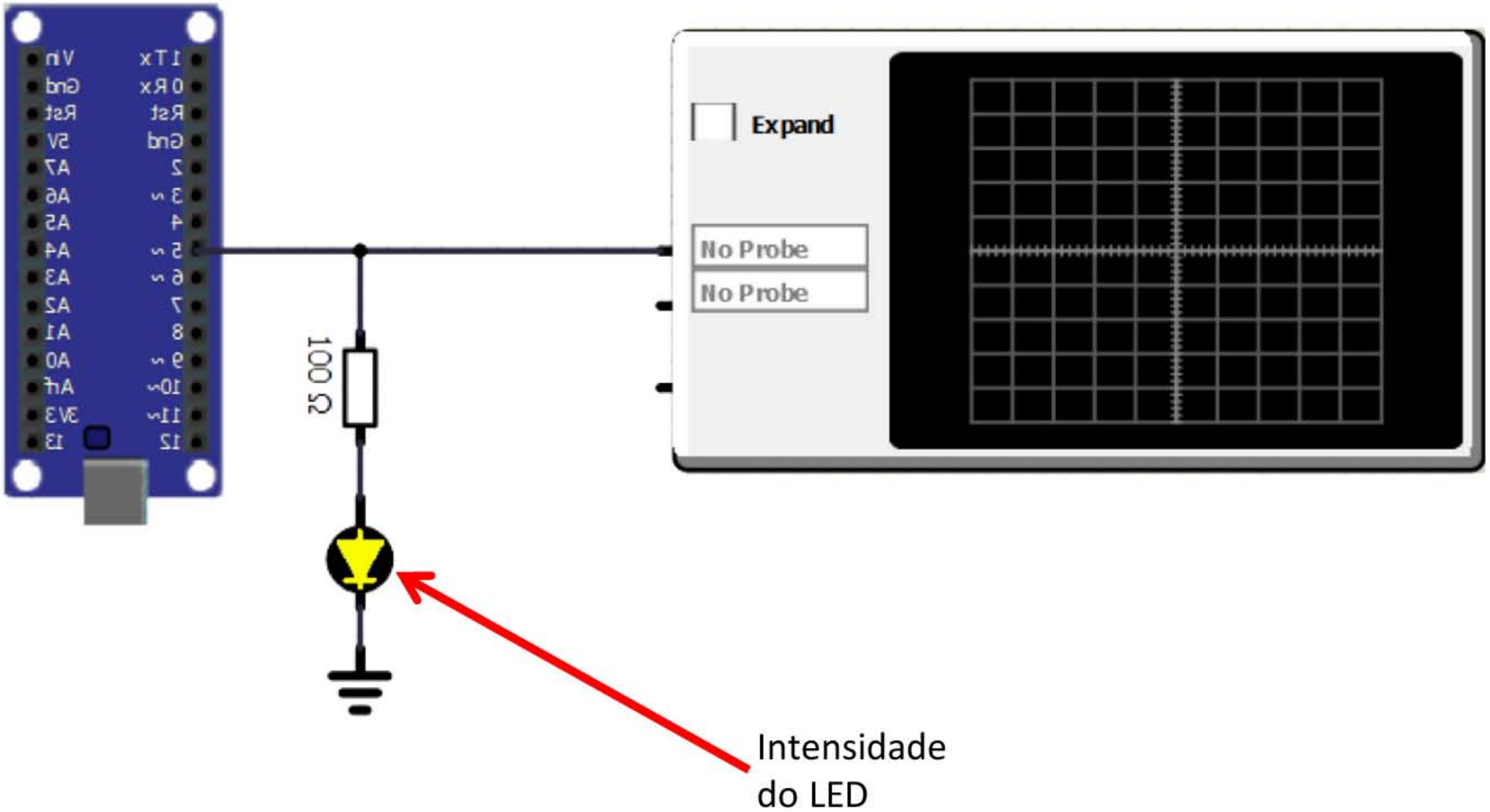
© Prof. Vargas

Esquema de teste 1



© Prof. Vargas

Esquema de teste 2



© Prof. Vargas

Programa Teste

```
/*-----------------------------------------------------------------
  Programa : PWM1
  Data     : 22/11/2020
  Autor    : Prof. Vargas
  Descrição: Gerar PWM nas portas digitais definidas para PWM nativo.
             O valor varia de 1 a 254, alterando a luminosidade do LED no
             PORT 10.
 -----------------------------------------------------------------*/
//Declaração de ports por meio de constante
const int Led1 = 10;
int i;

void setup()
{
   pinMode(Led1, OUTPUT);
}

void loop() //Ação normal repetitiva
{
   for(i=1;i<255;i++)  //Incrementa
   {
      analogWrite(Led1,i);
      delay(5); // Espera x segundos
   }
   for(i=254;i>0;i--)  //Decrementa
   {
      analogWrite(Led1,i);
      delay(5); // Espera x segundos
   }
}
```

© Prof. Vargas

Referências:

- **Souza, F.** - Usando as saídas PWM do Arduino - https://www.embarcados.com.br/pwm-do-arduino/ - 01/09/2019
- https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/ - 23/06/2020

© Prof. Vargas