

Componentes Eletronicos

Guia completo do Motor de Passo 28BYJ-48 + Driver ULN2003

Eletrogate 23 de julho de 2018 9 min

Nesse Guia completo do Motor de Passo 28BYJ-48 + Driver ULN2003, você verá :

Informações sobre o Motor de Passo 28BYJ-48:

Informações sobre o Módulo Driver ULN2003:

Motor de Passo - modos de operação:

Montagem do Motor e do Driver com Arduino:

Modo de Passo completo com alto torque (Full Step):

Aplicações interessantes para o Motor de Passo 28BYJ-48 + Driver ULN2003:

Sobre o Autor

Nesse Guia completo do Motor de Passo 28BYJ-48 + Driver ULN2003, você verá :

Introdução

Essa montagem, eu diria que é a mais simples para controle de um [motor](#) de passo (stepper motor). Esse motorzinho tem uma redução interna que aumenta bem o torque, apesar do pequeno tamanho. O módulo de controle com o chip ULN2003 é bem pequeno também, o que pode facilitar a montagem do circuito.

As vantagens são a simplicidade na montagem , Sketch pode ser bem enxuto, o uso de tensão de 5V para alimentar o motor, etc.

A desvantagens são a rotação baixa do motor (devido a redução na caixa de engrenagens) , o motor unipolar é menos eficiente do que o bipolar e o uso de quatro portas do Arduino. Alguns módulos de controle de motor de passo usam interfaces com duas portas apenas (por exemplo, com interface I2C).

Se você quiser saber mais sobre motores de passo, recomendo a leitura desse outro tutorial :

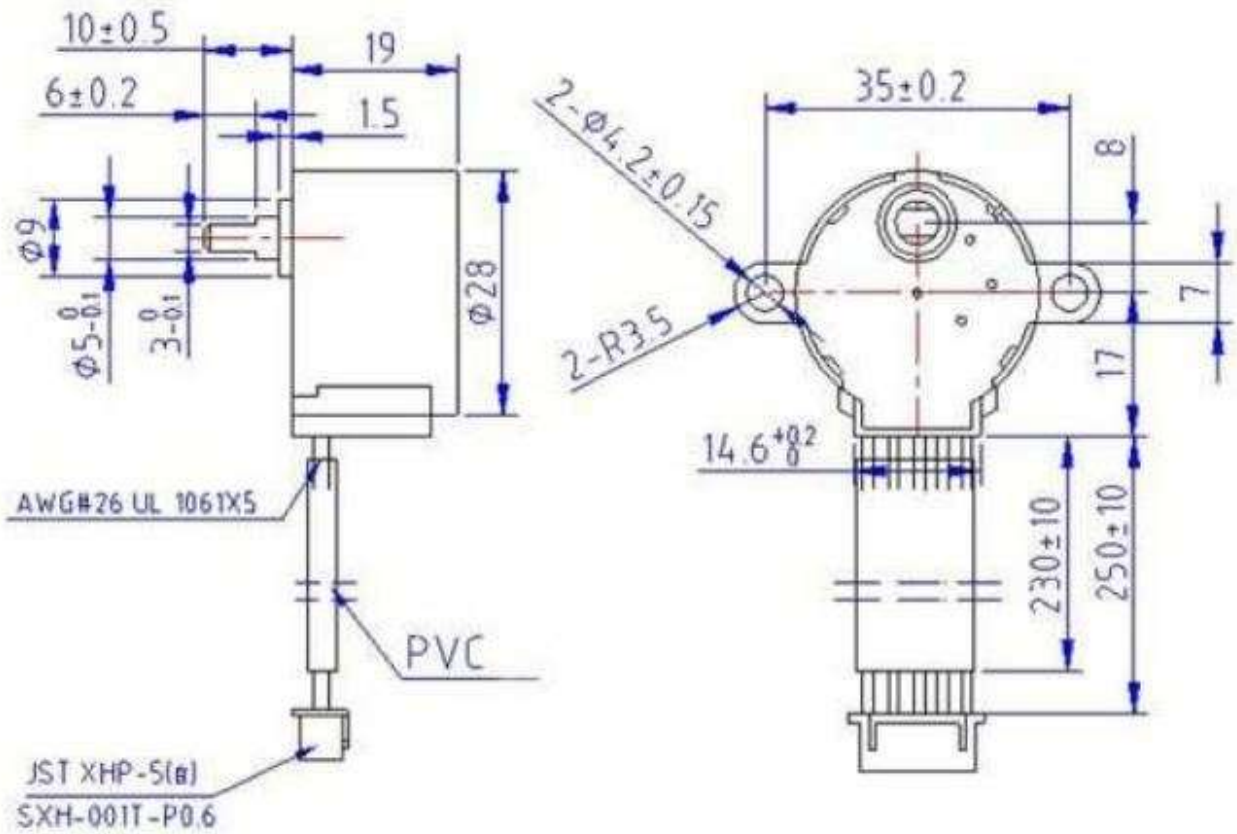
[Tudo sobre DRIVER A4988 e Motor de Passo > Usando o Arduino](#)

Informações sobre o Motor de Passo 28BYJ-48:

Esse é mais um motorzinho fabricado na China. Ele deve ser produzido em larga escala e por isso ele tem um preço bem acessível. Interessante , que ele pode ser alimentado com 5V e consome baixa corrente, o que facilita a montagem. Esse motor é Unipolar pois possui 4 enrolamentos que chamamos de fases. Em uma das pontas das fases, todas estão conectadas juntas. Portanto, esse motor não pode ser usado em Drivers para motores Bipolares (com duas fases somente).

Essas são as especificações do Motor 28BYJ-48:

[Motor28BYJ48Kiatronics](#)



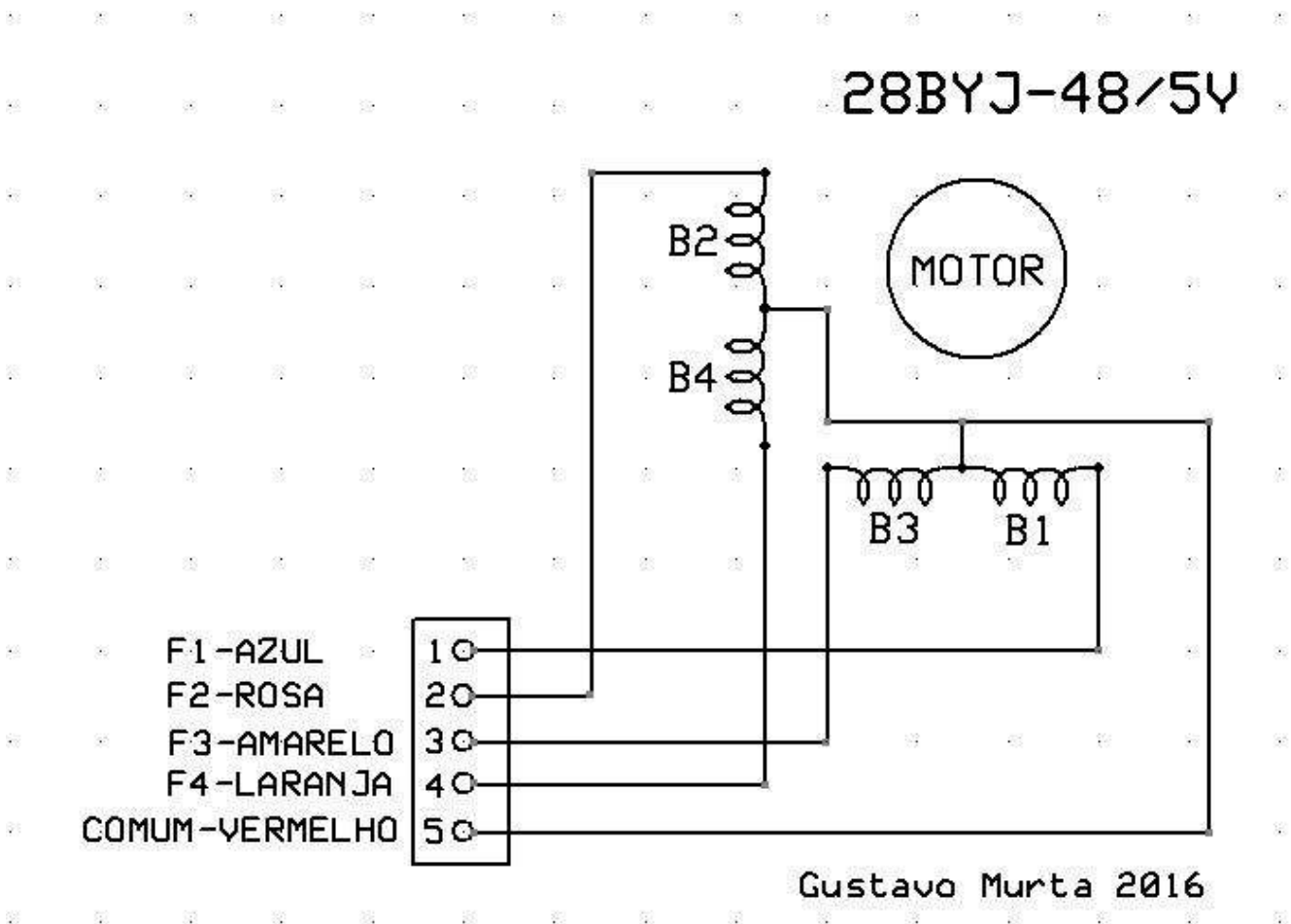
A distância entre os dois suportes de parafusos é de 35 mm. O diâmetro do motor é de 28 mm e a profundidade é de 19 mm. O diâmetro do eixo é 5mm, e ele é chanfrado.

- Tensão de operação : 5V CC
- Número de fases : 4
- Razão da variação de velocidade : 1/64 (mecanismo de redução)
- Angulo do passo : 5,625 graus => 64 passos/volta (360/64=5,625)
- Resistência CC : 50 ohms
- Frequência : 100 Hz
- Torque de tração: > 34,3 mN.m

OBS: a resistência medida nas bobinas do motor foi de 23 ohms. Assim podemos deduzir o consumo estático de corrente em cada bobina :

$I = V / R = 5V / 23 = 217 \text{ mA}$ aproximamente (a corrente medida foi um valor aproximado disso)

Esse é o diagrama do motor (desenhado por mim) :

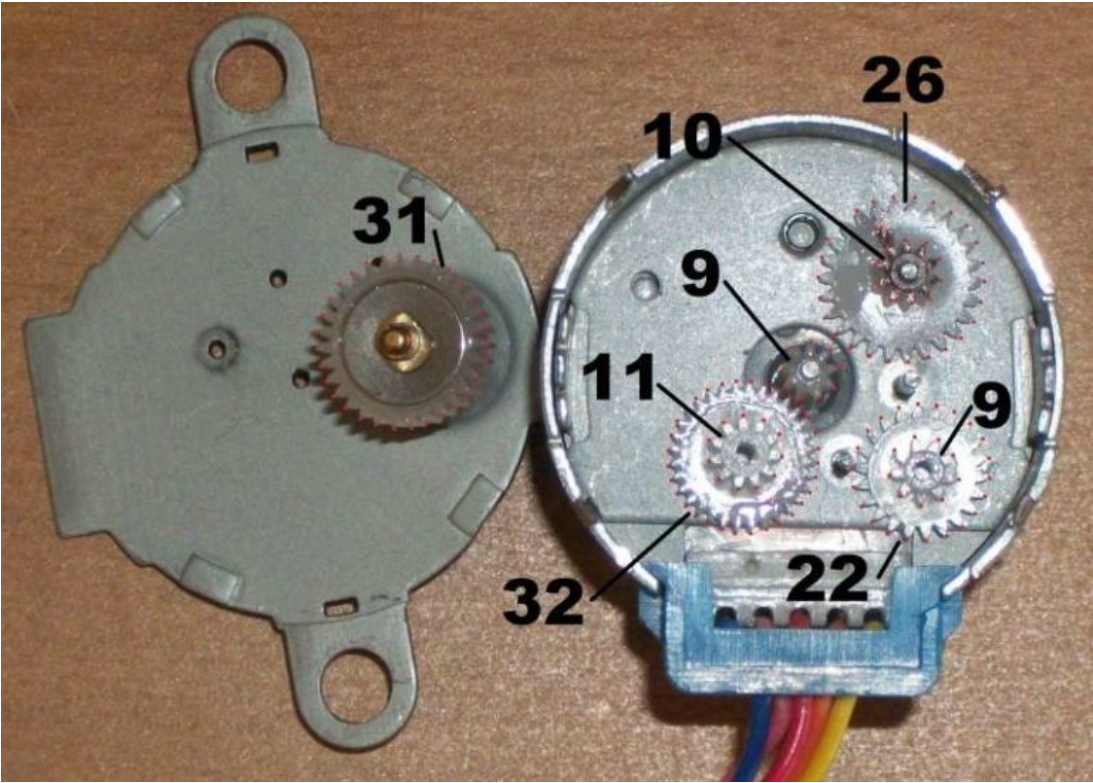


Para entender o mecanismo de redução do motor, veja esse excelente video :



No Forum do Arduino encontrei um t3pico onde desmontaram o motor, para entender como funciona a caixa de redu33o de engrenagens. Muito curioso !

Geared Stepper Motor



Como podem ver, a caixa tem v31rias engrenagens (31 dentes, 32 dentes, 26 dentes, 22 dentes, 11 dentes, 10 dentes, e mais duas com 9 dentes).

O pessoal fez alguns c31culos para determinar com maior precis33o qual era a redu33o.

$$(31 \cdot 32 \cdot 26 \cdot 22) / (11 \cdot 10 \cdot 9 \cdot 9) = 283712 / 4455 = 25792 / 405 = 63,68395...$$

Na especifica33o do fabricante, ele aproximou o valor para 64.

Para calcular o n31mero de passos do motor interno para girar uma volta do eixo externo (com redu33o) :

$$(64 \cdot 25792) / 405 = 4075,7728395..$$

Isto 33 , s33o necess31rios aproximadamente 4075 passos no motor de passo interno, para uma volta no eixo externo. Esse valor pode ter uma pequena varia33o devido 33s folgas nas engrenagens.

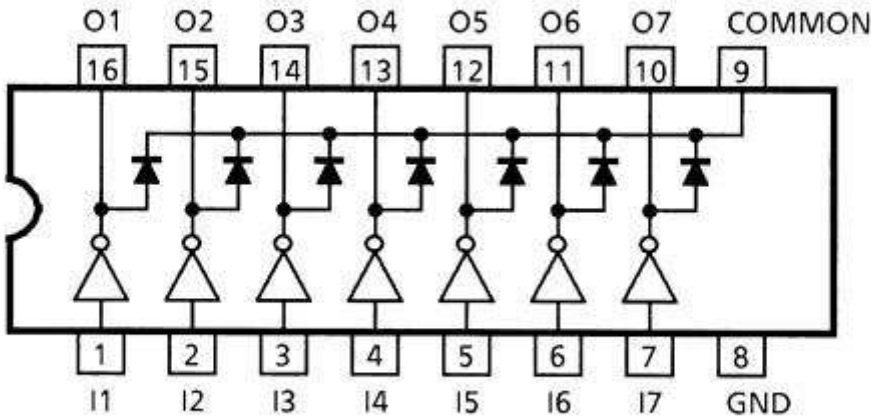
Informa33es sobre o M33dulo Driver ULN2003:

O unico chip no m33dulo 33 o ULN2003. Esse chip possui um conjunto de sete drivers de transistores Darlington que permitem o acionamento de cargas indutivas. Todas as sa33idas tem o coletor aberto e **diodos** de supress33o (Clamp) . Os transistores suportam tens33es de at33 50V e correntes de at33 500 mA. Todas as entradas IN1, IN2,IN3 e IN4 s33o

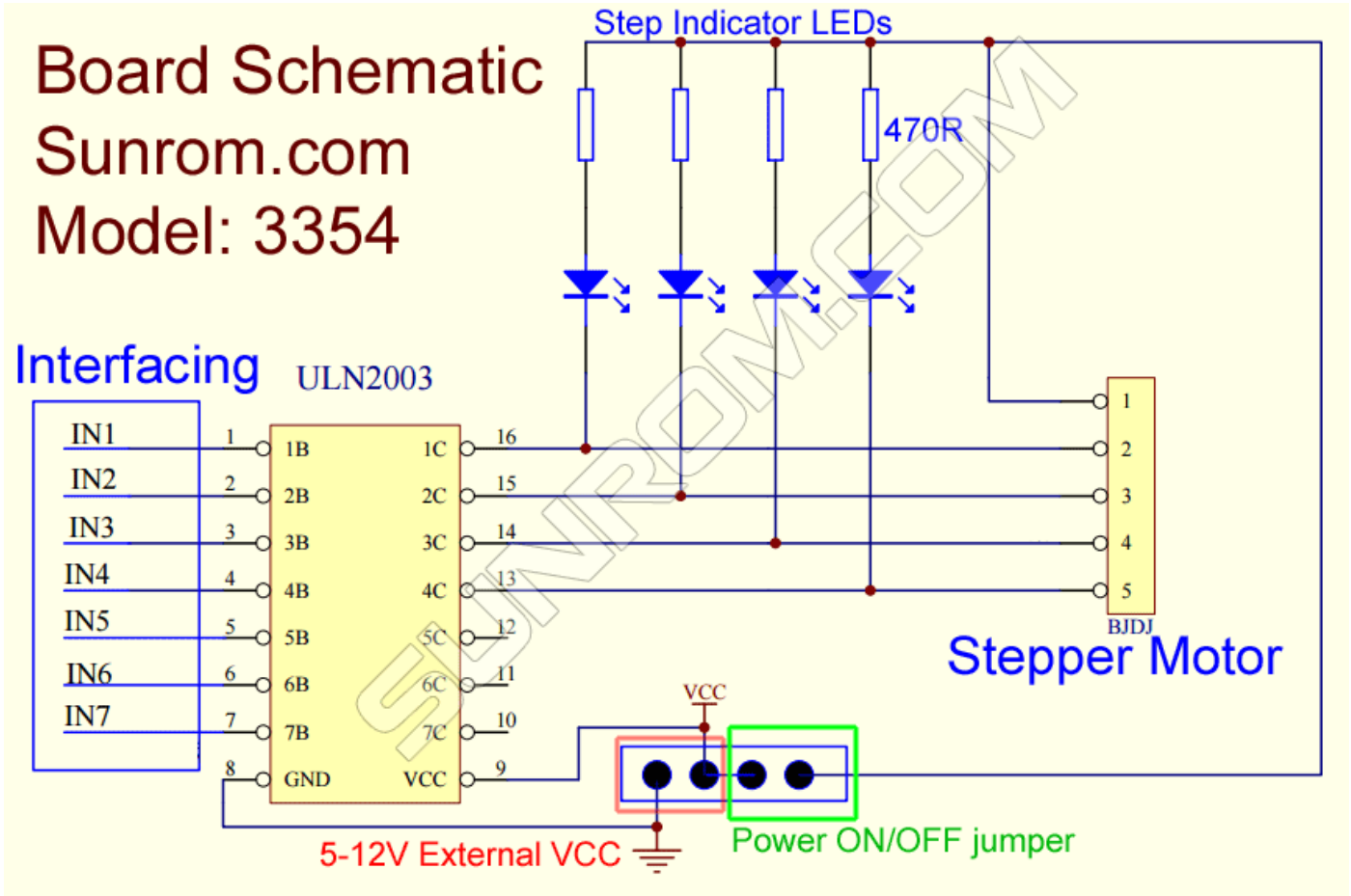
compatíveis com sinais TTL e CMOS, com limite de 5V. O pino comum tem que ser conectado na tensão de alimentação do motor. Nesse caso é conectado no 5V.

[Datasheet ULN2003APG](#)

Essa é disposição dos circuitos drivers no chip ULN2003APG:



Esse é o diagrama da placa de circuito do Módulo ULN2003APG :

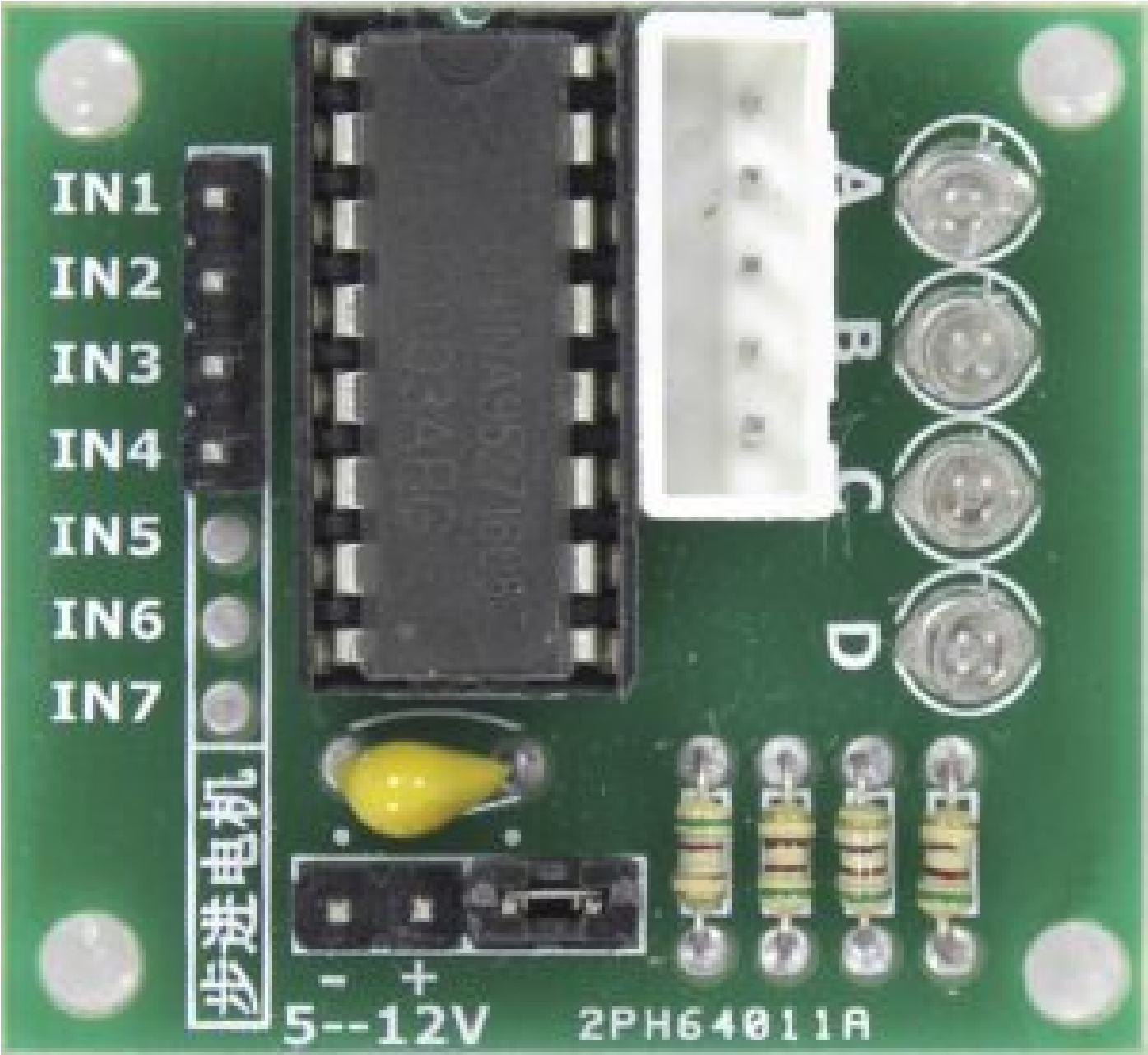


Os quatro leds vermelhos (A,B,C e D) são usados para indicar o acionamento de cada um dos drivers (fases do motor). Mesmo que o diagrama mostre sete entradas, somente quatro podem ser usadas e somente essas tem pinos no conector.

Muita atenção ao conectar a alimentação nos pinos.

O pino mais à esquerda (-) no módulo é o terra (veja foto abaixo). Esse terra tem que ser conectado ao terra da fonte e ao terra do Arduino. O pino (+) no caso do Motor 28BYJ-48, tem que ser conectado ao positivo de uma fonte de 5V (preferencialmente de 1 Ampére). Não recomendo que conecte no 5V do Arduino, pois poderá sobrecarregar o regulador de tensão do mesmo.

O jumper Power ON/OFF é usado para ligar ou desligar o motor.



Módulo ULN 2003 – foto Gustavo Murta

Motor de Passo - modos de operação:

Nesse tutorial o Motor de passo é Unipolar, isto é, ele possui quatro enrolamentos que chamamos de Fases.

Cada circuito driver do chip ULN2003A aciona uma das fases. E a ativação de cada driver é realizada pelas portas digitais do Arduino (portas D08, D09, D10 e D11) .

- F1 (azul) – D08
- F2 (rosa) – D09
- F3 (amarelo) – D10
- F4 (laranja) – D11

Para um motor de Passo Unipolar, temos alguns modos de operação. O modo Passo completo com alto torque (Full step), o modo Passo completo com baixo torque (Wave Step), o modo Meio Passo (half step) e Micro-passo (Micro stepping).

No caso do Micro passo, a corrente nos enrolamentos deve ser alterada. Como o nosso circuito não permite o controle da corrente, esse modo de operação não se aplica.

Para entender como as Fases são acionadas em cada caso, vejam essas cartas de tempo:

Obs: 0 (zero) = enrolamento desativado e 1 = enrolamento ativo.

Nas linhas estão a sequência dos passos e nas colunas estão as Fases.

Passo completo com alto torque (Full step):

Duas Fases são acionadas ao mesmo tempo.

Step	$\phi 4$	$\phi 3$	$\phi 2$	$\phi 1$
0	0	0	1	1
1	0	1	1	0
2	1	1	0	0
3	1	0	0	1

Passo completo com baixo torque (Wave Step):

Somente uma Fase acionada de cada vez.

Step	$\phi 4$	$\phi 3$	$\phi 2$	$\phi 1$
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	1	0	0	0

Meio Passo (half step):

Na sequência de oito passos, em alguns passos temos somente uma fase acionada e em outros passos, temos duas fases acionadas.

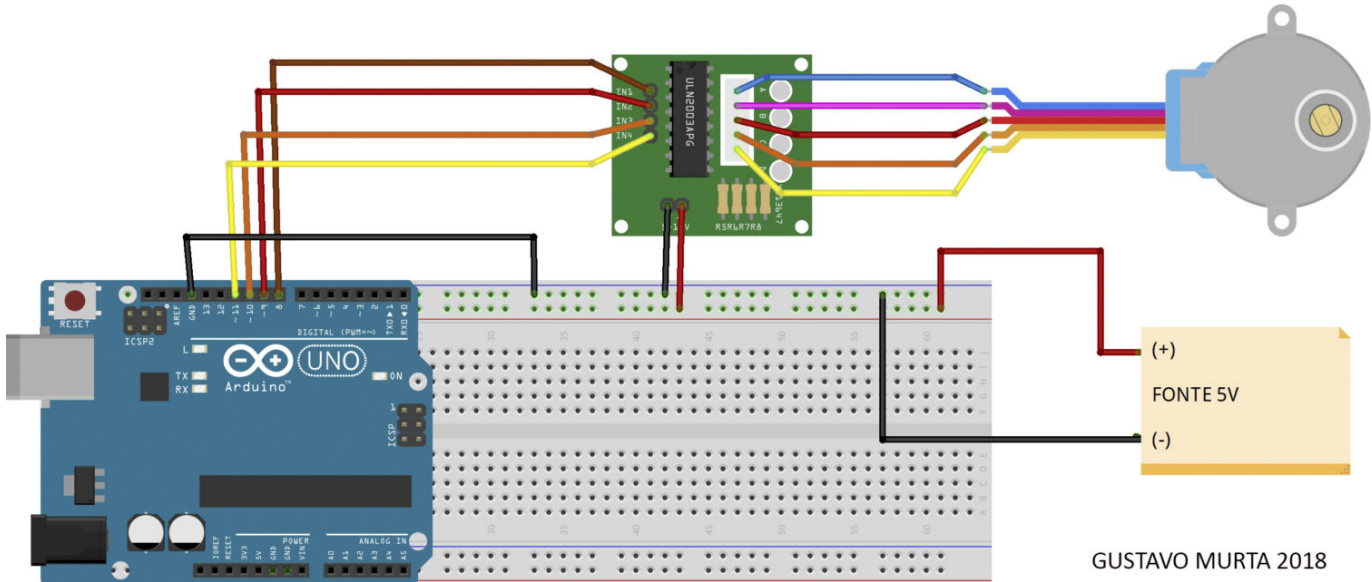
Step	$\phi 4$	$\phi 3$	$\phi 2$	$\phi 1$
0	0	0	0	1
1	0	0	1	1
2	0	0	1	0
3	0	1	1	0
4	0	1	0	0
5	1	1	0	0
6	1	0	0	0
7	1	0	0	1

Referência : [Stepper_Motors_2011.pdf](#)

Montagem do Motor e do Driver com Arduino:

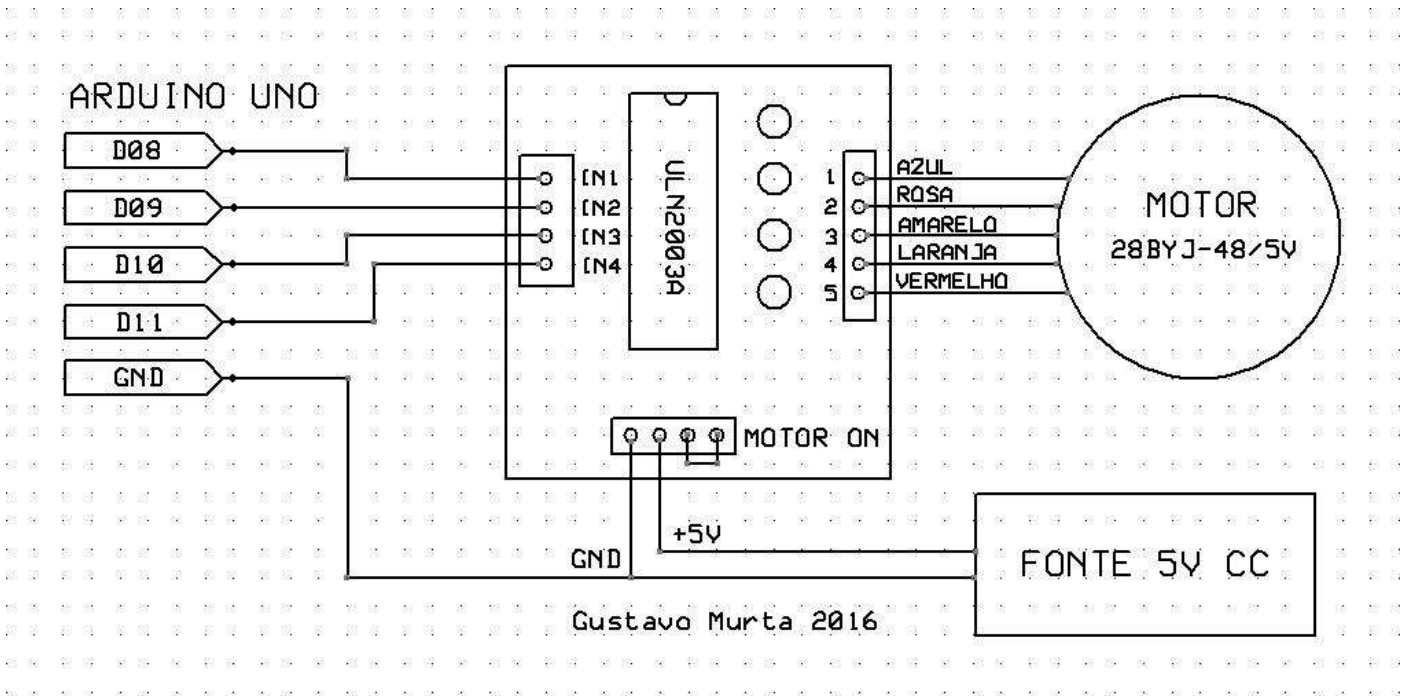
Esse é o diagrama da montagem do circuito com Arduino.

Obs: como já informei, recomendo que use uma fonte externa de 5V para alimentar o Driver + Motor de passo.



GUSTAVO MURTA 2018

Esse é o diagrama eletrônico do mesmo circuito:



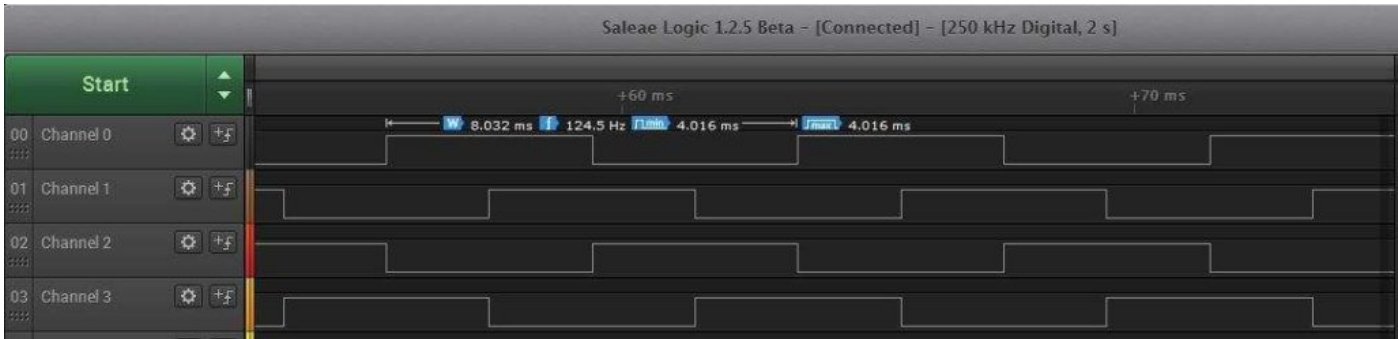
Modo de Passo completo com alto torque (Full Step):

No Modo de Passo completo com alto torque, para cada passo duas fases são ativadas simultaneamente. Esse modo é o mais usado, pois como o nome já diz, tem mais torque. O consumo de corrente é mais alto, devido à ativação das duas fases.

Um dos motivos para a escolha desse Módulo driver ULN2003 é que ele possui quatro leds que permitem a indicação da ativação das fases do motor. Para fins didáticos, isso é muito útil.

Essa é a Carta de Tempo do Modo Passo completo com alto torque (FullStep):

Veja que para cada passo, duas Fases (Channel) são ativadas ao mesmo tempo.



Para melhor visualizar o avanço dos Leds piscando de acordo com a ativação das Fases, altere a variável atraso-fase para 500 milisegundos :

```
int atraso_fase = 500 ;
```

Fica visível que dois Leds piscam ao tempo.

Verificando os valores dos Bytes das matrizes do programa : AHO e HOR , percebe-se que dois bits são ativados ao mesmo tempo. E é claro, uma sequência de bytes é inversa da outra.

Full_step.ino

```
1. // Controle de Motor de Passo - Modo Passo Completo alto torque (Full step)
2. // Blog Eletrogate - https://blog.eletrogate.com/guia-completo-do-motor-de-pas
3. // Baseado em http://www.elecrow.com/wiki/index.php?title=ULN2003_Stepper_Mot
4. // Motor 28BYJ48/5V com Módulo ULN20023 - Arduino Nano / IDE 1.8.5
5. // Uma volta no eixo = 4075 pulsos / 512 x 8 = 4096
6. // Gustavo Murta 23/jul/2018
7.
8. byte HOR[4] = {0x09,0x03,0x06,0x0C}; // Matriz dos bytes das Fases do Motor
9. byte AHO[4] = {0x0C,0x06,0x03,0x09}; // Matriz dos bytes das Fases do Motor
10. int atraso_fase = 2 ; // Intervalo de tempo entre as fases e
11. int intervalo = 1000 ; // Intervalo de tempo entre os movimen
12.
13. void Motor_AHO() // Movimento no sentido anti-horário
14. {
15.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
16.
17.         for(int j = 0; j < 4; j++) // incrementa o contador j de 0 a 3
18.         {
19.             PORTB = AHO[j]; // Carrega bytes da Matriz AHO na Porta B
20.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
21.         }
22. }
23.
24. void Motor_HOR() // Movimento no sentido horário
25. {
26.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
27.
28.         for(int j = 0; j < 4; j++) // incrementa o contador j de 0 a 3
29.         {
30.             PORTB = HOR[j]; // Carrega bytes da Matriz HOR na Porta B
31.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
32.         }
33. }
34.
35. void setup()
36. {
37.     DDRB = 0x0F; // Configura Portas D08,D09,D10 e D11 como saída
38.     PORTB = 0x00; // Reset dos bits da Porta B (D08 a D15)
39. }
40.
41. void loop()
42. {
43.     Motor_HOR(); // Gira motor no sentido Horário
44.     delay (intervalo); // Atraso em milisegundos
45.     Motor_AHO(); // Gira motor no sentido Anti-Horário
46.     delay (intervalo); // Atraso em milisegundos
47. }
```

Referências:
http://www.elecrow.com/wiki/index.php?title=ULN2003_Stepper_Motor_D...

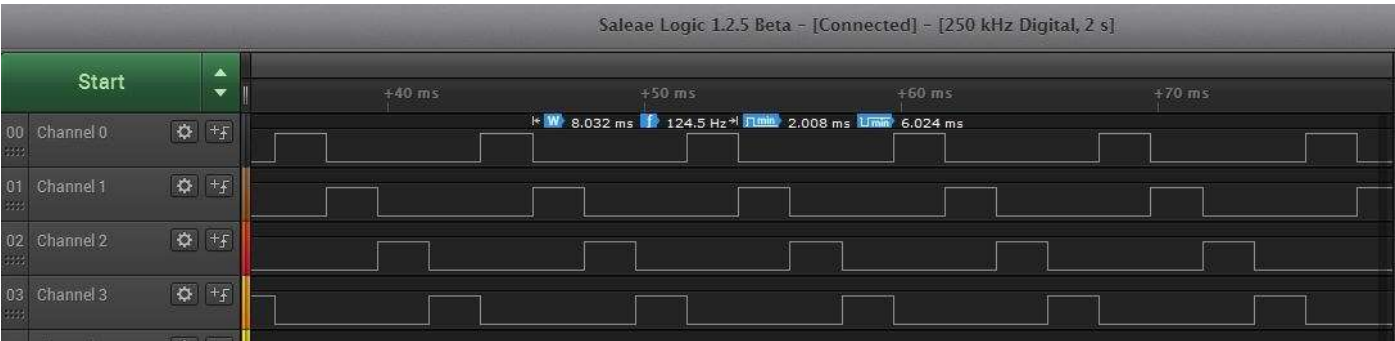
Modo de Passo completo com baixo torque (Wave Step):

Como já foi dito, no modo Passo completo com baixo torque (Wave step) somente uma fase é acionada de cada vez, portanto o torque do motor é menor do que nos outros modos. Esse modo é pouco usado exatamente por isso. O rendimento do motor é menor, mas tem-se como vantagem um menor consumo de corrente,já que somente uma fase é ativada em cada passo.

Para melhor visualizar o avanço dos Leds piscando de acordo com a ativação das Fases, altere a variável atraso-fase para 500 milissegundos :

```
int atraso_fase = 500 ;
```

Desse modo poderá perceber que os Leds piscam um de cada vez , confirmando a sequência da Carta de Tempo:



Importante ressaltar que nas matrizes do programa : AHO e HOR , estão carregados os bytes de ativação das fases, de acordo com a tabela anexada no tópico anterior. Uma sequência de bytes é inversa da outra.

[wave_step.ino](#)

```
1. // Controle de Motor de Passo - Modo Passo Completo baixo torque (Wave step)
2. // Blog Eletrogate - https://blog.eletrogate.com/guia-completo-do-motor-de-pas
3. // Baseado em http://www.elecrow.com/wiki/index.php?title=ULN2003\_Stepper\_Mot
4. // Motor 28BYJ48/5V com Módulo ULN20023 - Arduino Nano / IDE 1.8.5
5. // Uma volta no eixo = 4075 pulsos / 512 x 8 = 4096
6. // Gustavo Murta 23/julho/2018
7.
8. byte HOR[4] = {0x01,0x02,0x04,0x08}; // Matriz dos bytes das Fases do Motor
9. byte AHO[4] = {0x08,0x04,0x02,0x01}; // Matriz dos bytes das Fases do Motor
10. int atraso_fase = 2 ; // Intervalo de tempo entre as fases e
11. int intervalo = 1000 ; // Intervalo de tempo entre os movimen
12.
13. void Motor_AHO() // Movimento no sentido anti-horário
14. {
15.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
16.
17.         for(int j = 0; j < 4; j++) // incrementa o contador j de 0 a 3
18.         {
19.             PORTB = AHO[j]; // Carrega bytes da Matriz AHO na Porta B
20.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
21.         }
22. }
23.
24. void Motor_HOR() // Movimento no sentido horário
25. {
26.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
27.
28.         for(int j = 0; j < 4; j++) // incrementa o contador j de 0 a 3
29.         {
30.             PORTB = HOR[j]; // Carrega bytes da Matriz HOR na Porta B
31.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
32.         }
33. }
34.
35. void setup()
36. {
37.     DDRB = 0x0F; // Configura Portas D08,D09,D10 e D11 como saída
38.     PORTB = 0x00; // Reset dos bits da Porta B (D08 a D15)
39. }
40.
41. void loop()
42. {
43.     Motor_HOR(); // Gira motor no sentido Horário
44.     delay (intervalo); // Atraso em milissegundos
45.     Motor_AHO(); // Gira motor no sentido Anti-Horário
46.     delay (intervalo); // Atraso em milissegundos
47. }
```

Modo de Meio Passo (Half Step):

O Modo Meio Passo é uma mesclagem dos outros dois modos, Full Step e Wave Step.

Esse modo é usado quando o movimento do motor precisa ser delicado e preciso e o torque não é importante.

O motor interno precisa de 64 passos para uma revolução (volta), no modo Half Step o número de passos dobra para 128 passos !

Por isso é chamado de Meio Passo. O efeito Meio passo é conseguido intercalando um passo (com duas fases do Motor ativadas simultaneamente) com outro passo (ativando somente uma fase), obedecendo uma sequência própria para isso. Devido à essa alternância, o torque e o consumo de corrente variam de acordo com o passo. Com mais fases ativas, terá maior torque e maior consumo de corrente.

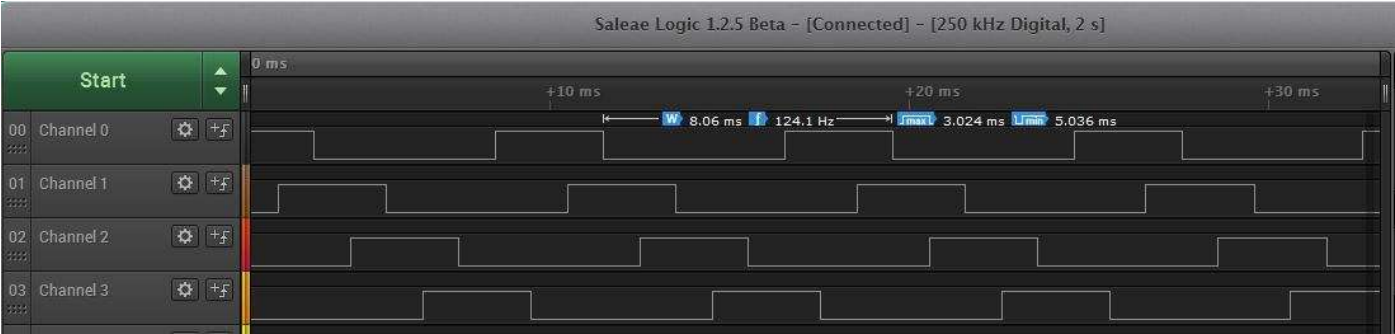
O interessante nesse modo é que as matrizes dos Bytes AHO e HOR , tem oito valores e não somente quatro como nos outros modos. Analisando os Bytes, dá para perceber que as matrizes dos outros modos foram agrupadas.

Half_step.ino

```
1. // Controle de Motor de Passo - Modo Meio Passo (Half step)
2. // Blog Eletrogate - https://blog.eletrogate.com/guia-completo-do-motor-de-pas
3. // Baseado em http://www.elecrow.com/wiki/index.php?title=ULN2003_Stepper_Mot
4. // Motor 28BYJ48/5V com Módulo ULN20023 - Arduino Nano / IDE 1.8.5
5. // Uma volta no eixo = 4075 pulsos / 512 x 8 = 4096
6. // Gustavo Murta 23/jul/2018
7.
8. byte HOR[8] = {0x09,0x01,0x03,0x02,0x06,0x04,0x0c,0x08}; // Matriz dos byte
9. byte AHO[8] = {0x08,0x0c,0x04,0x06,0x02,0x03,0x01,0x09}; // Matriz dos byte
10. int atraso_fase = 1 ; // Intervalo de te
11. int intervalo = 1000 ; // Intervalo de te
12.
13. void Motor_AHO() // Movimento no sentido anti-horário
14. {
15.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
16.     {
17.         for(int j = 0; j < 8; j++) // incrementa o contador j de 0 a 7
18.         {
19.             PORTB = AHO[j]; // Carrega bytes da Matriz AHO na Porta B
20.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
21.         }
22.     }
23.
24. void Motor_HOR() // Movimento no sentido horário
25. {
26.     for(int i = 0; i < 512; i++) // incrementa o contador i de 0 a 511 - um
27.     {
28.         for(int j = 0; j < 8; j++) // incrementa o contador j de 0 a 7
29.         {
30.             PORTB = HOR[j]; // Carrega bytes da Matriz HOR na Porta B
31.             delay (atraso_fase); // Atraso de tempo entre as fases em milis
32.         }
33.     }
34.
35. void setup()
36. {
37.     DDRB = 0x0F; // Configura Portas D08,D09,D10 e D11 como saída
38.     PORTB = 0x00; // Reset dos bits da Porta B (D08 a D15)
39. }
40.
41. void loop()
42. {
43.     Motor_HOR(); // Gira motor no sentido Horário
44.     delay (intervalo); // Atraso em milisegundos
45.     Motor_AHO(); // Gira motor no sentido Anti-Horário
46.     delay (intervalo); // Atraso em milisegundos
47. }
```

Essa é a Carta de Tempo do Modo Meio Passo (Half Step):

Em alguns passos, tem duas fases ativadas e em outros tem somente uma fase ativada .



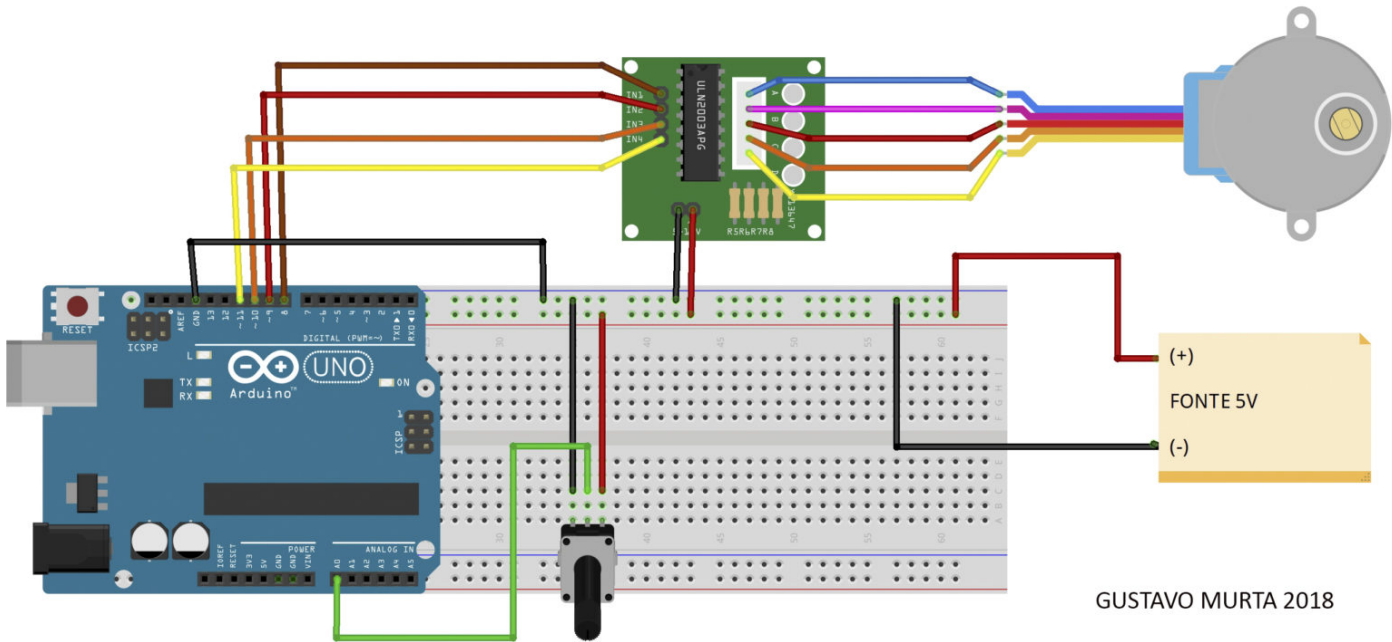
Bibliotecas para Motor de passo:

Existem várias bibliotecas para o controle de motor de passo com o Arduino. As mais comuns são essas:

- [Biblioteca Stepper](#)
- [Biblioteca AccelStepper](#)
- [Biblioteca CustomStepper](#)

Exemplo de Skecth com a Biblioteca AccelStepper. Gire o potenciômetro, e o motor irá acompanhar o movimento do mesmo (Giro total de 180 graus do eixo do motor). Use um potenciômetro de 10 K ohms.

Diagrama do circuito :



AccelStepperPOT.ino

```
1. // Controle de Motor de Passo com Potenciometro
2. // Blog Eletrogate - https://blog.eletrogate.com/guia-completo-do-motor-de-pas
3. // Baseado em http://www.airspayce.com/mikem/arduino/AccelStepper/Proportional
4. // Motor 28BYJ48/5V com Modulo ULN20023 - Arduino UNO / IDE 1.8.5
5. // Uma volta no eixo = 4075 pulsos
6. // Gustavo Murta 02/jago/2018
7.
8. #include <AccelStepper.h> // bibliote
9.
10. #define ANALOG_PIN A0 // pino A0
11.
12. AccelStepper motorPasso (AccelStepper::FULL4WIRE, 8, 10, 9, 11); // Passo co
13.
14. void setup()
15. {
16.     motorPasso.setMaxSpeed(500); // maxima ve
17. }
18.
19. void loop()
20. {
21.     int analog_in = analogRead(ANALOG_PIN); // lendo a
22.     motorPasso.moveTo(analog_in); // gira o e
23.     motorPasso.setSpeed(100); // veloci
24.     motorPasso.runSpeedToPosition(); // gira o e
25. }
```

Aplicações interessantes para o Motor de Passo 28BYJ-48 + Driver ULN2003:

Esse motor de passo permite movimentos precisos e com baixa rotação. Essas são algumas aplicações interessantes voltadas para fotografia :

[ARDUINO TIME-LAPSE PANORAMA CONTROLLER](#)

[Motion Control Rig for Time-Lapse Photography](#)

Quer saber mais sobre **Arduino** ?

[Placa Arduino](#) | [Arduino](#) | [Case Arduino](#) | [Placas Arduino](#) | [Kit Arduino](#) | [Sensores Arduino](#)

[Se tiver alguma dúvida, deixe um comentário!](#)


Sobre o Autor

**José Gustavo
Abreu Murta**

Consultor e Projetista de Sistemas Embarcados. Técnico em eletrônica, formado em Curso superior de TPD, pós-graduado em Marketing. Trabalhou por muitos anos na IBM na área de manutenção de computadores de grande porte. Aposentou-se, podendo curtir o que mais gosta : estudar e ensinar Tecnologia. Hobista em eletrônica desde 1976. Gosta muito de Fotografia e Observação de aves.

Eletrogate 23 de julho de 2018


A Eletrogate é uma loja virtual de componentes eletrônicos do Brasil e possui diversos produtos relacionados à Arduino, Automação, Robótica e Eletrônica em geral.



Muito obrigado!

^ | v

• Responder • Compartilhar ›




marcos machado • um ano atrás

Preciso controlar VÁRIOS MOTORES DE PASSO (uns 40) em um RaspBerry PI, penso em usar as portas I2C (3 e 5) com extensores de portas. Como fazer a interface entre o RaspBerry PI e a controladora dos motores por I2C?
(motor de passo 28byj-48 5v/12v | controladora ULN2003 - 4 fase)

^ | v

• Responder • Compartilhar ›




Jose Gustavo Abreu Murta Mod ➔ marcos machado • um ano atrás

Boa tarde Marcos,
Seu projeto é ambicioso. Esse driver de motor utiliza 4 portas para cada motor. Então teria que usar 160 portas! Recomendo o uso de outro driver e de outro motor, que utilize menos portas.
Se realmente quiser usar 160 portas, pode tentar com o extensor de portas MCP23017 (16-Bit I/O Expander with Serial Interface)
<http://ww1.microchip.com/do...>
Atenciosamente,
Gustavo Murta

^ | v

• Responder • Compartilhar ›




Syllas Renato • um ano atrás

Olá Gustavo, tudo bem?
Estou projetando um compensador PID para controle de posição deste motor e gostaria de saber se você poderia me informar as constantes K e Tau do mesmo. Se puder me ajudar, com certeza me fará economizar um certo tempo e trabalho calculando experimentalmente.
Desde já, muitíssimo obrigado!

^ | v

• Responder • Compartilhar ›




Jose Gustavo Abreu Murta Mod ➔ Syllas Renato • um ano atrás

Boa noite Syllas,
Esse tutorial sobre PID, talvez possa ser útil.
<http://www.electronoobs.com...>
Atenciosamente,
Gustavo Murta

^ | v

• Responder • Compartilhar ›




Syllas Renato ➔ Jose Gustavo Abreu Murta • um ano atrás

Muito Obrigado Jose Gustavo, certamente será de grande proveito.

^ | v

• Responder • Compartilhar ›




Elson Alexandre • 2 anos atrás

Bom dia... Eu conseguiria substituir um motor de passo automotivo por esses para controlar a marcha lenta do veículo?

^ | v

• Responder • Compartilhar ›




Jose Gustavo Abreu Murta Mod ➔ Elson Alexandre • 2 anos atrás

Boa tarde Elson,
Você tem que saber a força (torque) que o motor de passo do seu carro precisa para controlar a marcha lenta.
E comparar com o torque do motor desse tutorial. Ou então faça você mesmo a experiência.
Mas atenção! O motor desse tutorial é unipolar e o motor de passo do seu carro deve ser bipolar. Os circuitos são diferentes:
<https://blog.eletrogate.com...>
Atenciosamente,
Gustavo Murta

^ | v

• Responder • Compartilhar ›




Gustavo Menin • 2 anos atrás

Olá boa tarde, testei seu sketch (full_step) usando o mesmo motor o mesmo modulo porém com arduino Mega, não acontece nada nem os Leds não acendem poderia me ajudar ?

^ | v



• Responder • Compartilhar ›




Jose Gustavo Abreu Murta Mod ➔ Gustavo Menin • 2 anos atrás



Bom dia Gustavo,
Você usou os mesmos pinos do tutorial ? (D8 a D11)


Qual fonte de alimentação 5V usou? O módulo Driver do Motor precisa ser alimentado com 5V.
Certifique-se que o Jumper MOTOR ON esta conectado.
Atenciosamente,
Gustavo Murta

 |  • Responder • Compartilhar ›



**Gustavo Menin** ➔ Jose Gustavo Abreu Murta • 2 anos atrás


Eu usei os mesmos pinos sim, porém não tenho nenhuma fonte externa para alimentação então usei o próprio arduino

 |  • Responder • Compartilhar ›



**Jose Gustavo Abreu Murta** Mod ➔ Gustavo Menin • 2 anos atrás


Boa noite,
Verifique todas as ligações e o jumper. Meça a tensão 5V quando tenta fazer o teste do motor.
Me informe o valor medido.Como eu informei no tutorial, recomendo o uso de uma fonte externa de 5 V. O motor pode consumir mais corrente do que o Arduino possa fornecer.
Atenciosamente,
Gustavo Murta

 |  • Responder • Compartilhar ›



**Roger** • 2 anos atrás


Olá, Gustavo! Gostaria de perguntá-lo se fez o diagrama do circuito no Fritzing. Como consigo incluir o micromotor e o driver ULN2003 na biblioteca? Grato!

 |  • Responder • Compartilhar ›

**Jose Gustavo Abreu Murta** Mod ➔ Roger • 2 anos atrás

Boa tarde Roger,
O diagrama do Tutorial foi feito no Fritzing. Você esta me perguntando como incluí os desenhos na Biblioteca do Fritzing?
Não me lembro se eu incluí os desenhos na Biblioteca. Mas você pode copiar de um outro diagrama para o seu.
As instruções do Fritzing estão aqui:
<https://fritzing.org/learning/>
Atenciosamente,
Gustavo Murta

 |  • Responder • Compartilhar ›

**Raphael Passos** • 2 anos atrás



Boa tarde Jose, como vai ?


Então....estou precisando executar um dos exemplos mostrados {AccelStepper) mas está dando ERRO: `FULL4WIRE` is not a member of `AccelStepper` quando vou rodar no IDE.

Já procurei em muitos lugares e esse exemplo é o que preciso, mover o motor de passo acompanhando a posição do pot.

Teria uma outra solução ?

Valeu.

 |  • Responder • Compartilhar ›

**Jose Gustavo Abreu Murta** Mod ➔ Raphael Passos • 2 anos atrás

Boa noite Raphael,

Testei agora o sketch " AccelStepperPOT.ino" na Arduino IDE 1.8.9 e compilou com sucesso.



Procedimento para a instalação da Biblioteca:

Clique em :

Sketch > Incluir Biblioteca > Gerenciar Bibliotecas

Após abrir a janela do Gerenciador de Biblioteca, refine a busca digitando **accelstepper**. Na biblioteca **accelstepper**, clique em **More Info** e depois em Instalar. Após alguns segundos, ela será automaticamente instalada. Lembre-se que o seu computador precisa estar conectado na internet, para poder baixar a biblioteca. Após a instalação da Biblioteca, é necessário que feche e abra novamente o programa Arduino IDE.

Atenciosamente,
Gustavo Murta

 |  • Responder • Compartilhar ›



Gustavo Ferreira da Silva • 2 anos atrás

Boa noite, preciso programar com este mesmo motor de passo um sistema de seguidor solar, porém não encontro modelos de programa nessa situação, apenas modelos de programação de servo motores, porém pro peso da minha placa não daria.}

Alguém poderia me ajudar na síntese da programação para um seguidor solar de 4 LDR's nesse motor de passo 28BYJ-48?

^ | v • Responder • Compartilhar ›



Jose Gustavo Abreu Murta Mod ➔ Gustavo Ferreira da Silva • 2 anos atrás

Boa noite Gustavo,

Recomendo que use um Arduino Mega, pois tem um número maior de portas digitais.

Para cada motor precisará 4 portas, portanto no total 16 portas. O único projeto pronto com motores de passo, que eu encontrei foi esse. Terá que adaptar o seu projeto.

<http://labdegaragem.com/for...>

Veja esses projetos seguidores do Sol:

<http://labdegaragem.com/for...>

<https://www.portal-energia....>

<https://www.instructables.c...>

<http://www.cerebralmeltdown...>



ver mais

^ | v • Responder • Compartilhar ›



Paulo ➔ Jose Gustavo Abreu Murta • um ano atrás

Siga os conceitos do Great Scott (



e use a biblioteca stepper para controlar seus motores, não tem segredo.

^ | v • Responder • Compartilhar ›



Kirara • 2 anos atrás

Como eu descubro a resistência do motor?

^ | v • Responder • Compartilhar ›



Jose Gustavo Abreu Murta Mod ➔ Kirara • 2 anos atrás

Bom dia Kirara,

Todas as especificações do motor estão no Datasheet (que consta no tutorial):

Motor28BYJ48Kiatronics

DC resistance = 50Ω ± 7%(25°C)

Atenciosamente,

Gustavo Murta

^ | v • Responder • Compartilhar ›



Kirara ➔ Jose Gustavo Abreu Murta • 2 anos atrás

No exemplo vc diz que deu 23ohm, então se eu fizer $5/50=0.1A$ a corrente do motor seria isso ou seria só de uma fase?

^ | v • Responder • Compartilhar ›



Jose Gustavo Abreu Murta Mod ➔ Kirara • 2 anos atrás

Kirara,
Como eu informei no tutorial, na especificação consta 50 ohms cada enrolamento (fase).
Mas o que eu medi no meu motor foi 23 ohms por fase.
Recomendo que meça a resistência do seu motor com um ohmímetro.
Atenciosamente,
Gustavo Murta
^ | v • Responder • Compartilhar ›



Kirara ➔ Jose Gustavo Abreu Murta • 2 anos atrás

Ok, muito obrigado, uma última dúvida, se eu colocar uma fonte de 9v para alimentar o arduino, no pino vin sairá 9v ou 5v?

^ | v • Responder • Compartilhar ›



Jose Gustavo Abreu Murta Mod ➔ Kirara • 2 anos atrás

Kirara,
Se conectar uma fonte de 9 V no Arduino UNO, o pino VIN deverá ter a tensão aproximada de entrada = 9 V (aproximadamente, pois tem um diodo entre o conector de alimentação e o pino VIN).
Atenciosamente,
Gustavo Murta
^ | v • Responder • Compartilhar ›



Ana Caroline Lana • 2 anos atrás

Preciso das dimensões desse chanfro. Pode fornecer?

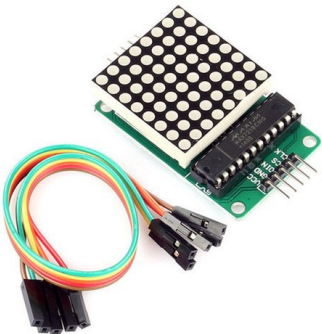
^ | v • Responder • Compartilhar ›

[COMPONENTES ELETRONICOS](#)

Tutorial Matriz de LED no Arduino

*Eletrogate*22 de julho de 2021

[Você está precisando de um display para mostrar informações, mas não sabe qual escolher? Aprenda um pouco sobre a matriz de LED.](#)



[PROJETOS](#)

Previsão do Tempo com Arduino e Display LCD

*Eletrogate*19 de julho de 2021

[Você já pensou em ficar sabendo a previsão do tempo com seu arduino? Então este post é pra você.](#)

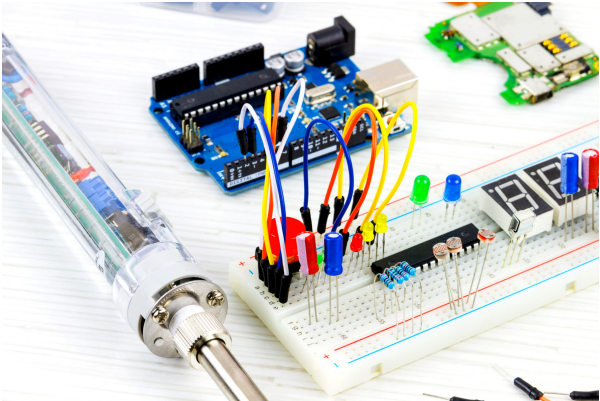


[COMPONENTES ELETRONICOS](#)

Expandindo as portas do Arduino

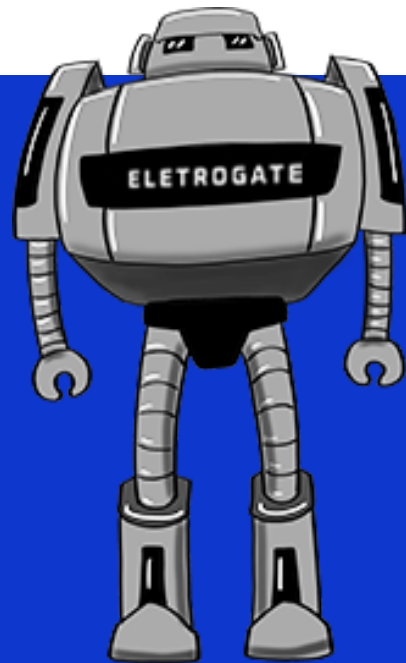
*Eletrogate*16 de julho de 2021

[Se você está atrás de uma solução para uma aplicação em Arduino que usa uma quantidade imensa portas, confira essa dica super... legal](#)



[AUTOMAÇÃO RESIDENCIAL](#)**Controle de Cargas por Adafruit IO***Eletrogate* 14 de julho de 2021

[Já pensou em controlar equipamentos da sua casa pela internet utilizando um arduino? Pois neste post você aprenderá como fazer... isso e ainda desenvolver uma plataforma personalizada para fazer o](#)



Cadastre-se e fique por dentro de novidades!

Nome *

Email *

☐ Eu concordo em receber comunicações.

Ao informar meus dados, eu concordo com a [Política de Privacidade](#).

Cadastrar**BLOG** ELETROGATE**Nos acompanhe**[Facebook](#)[Instagram](#)[Youtube](#)

© ELETROGATE 2021 - Todos os direitos reservados. Termos de uso e Política de privacidade.