

# LEARN ! DO ! ARDUINO.INFO.INFO

## SmallSteppers (/SmallSteppers)

Edit

2 (/SmallSteppers#discussion)

50 (/page/history/SmallSteppers)

... (/page/menu/SmallSteppers)

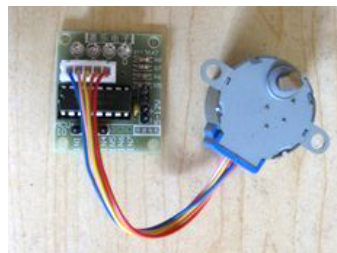
## Small Stepper Motor and Driver Board

[See this product here:](#)

[See 1-turn Arduino Sketch HERE:](#)

## MOTOR DETAILS

This is a 5v 28YBJ-48 Stepper Motor with Gear Reduction, so it has good torque for its size, but relatively slow motion. These motors/drivers are made by the millions for A/C units, fans, duct controls etc. which is why they are so inexpensive.



### 4 Phase 5 Wire Connection

- 100% Brand New
- Phase : 4
- Current : 160 mA per winding (320 mA in 4-step mode) Measured: 250mA stopped, 200 mA running fast
- Resistance : 31  $\Omega$  per coil winding (from Red wire to any coil) (Some 24-28 ohms)
- Voltage : 5V DC
- Step Angle (8-Step sequence: Internal Motor alone): 5.625° (64 steps per revolution)
- Step Angle (4-Step sequence: Internal Motor alone): 11.25° (32 steps per revolution)
- Gear Reduction ratio: 1 / 64 (Not really exact: probably 63.68395.:1 )
- SO: it takes (64\*64 = 4096 steps per output shaft revolution.. In 8-step sequence.
- SO: it takes (32\*64 = 2048 steps per output shaft revolution.. In 4-step sequence.
- NOTE: Arduino "Stepper Library" runs in 4-step mode
- No-Load Pull-Out Frequency : 800pps



Search

[HOME](#)

[ABOUT US!](#)

[YourDuinoShop](#)

[NEWSLETTERS](#)

[BLOG](#)

[THIS WIKI:](#)

[WHAT IS Arduino?](#)

[LEARNING Arduino:](#)

- [Arduino: WhatIsIt?](#)
- [Hands-On Learning](#)
- [Example Sketches](#)
- [Arduino Libraries](#)

[ARDUINO HOWTO:](#)

- [CABLES, WIRES and PINS](#)
- [SENSOR SHIELD](#)
- [ARDUINO POWER](#)
- [SERVO MOTORS](#)
- [WIRELESS nRF24L01](#)

- No-Load Pull-In Frequency : 500pps
- Pull-In Torque :  $\geq 78.4\text{mN.m}$
- Wiring Instruction : A (Blue), B (Pink), C (Yellow), D (Orange), E (Red, Mid-Point)
- Weight : 30g

See wiring diagram, following...

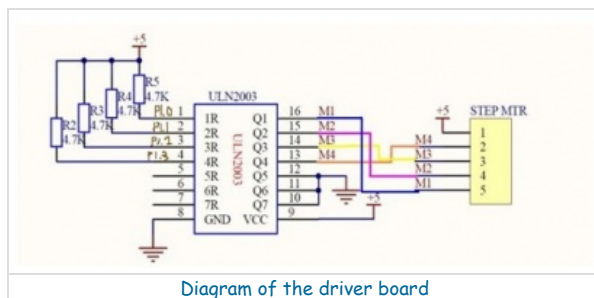
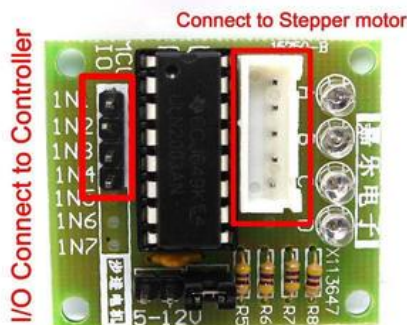
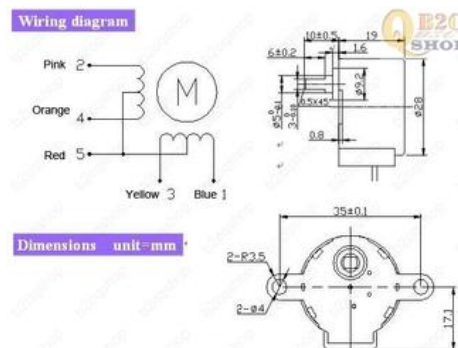


Diagram of the driver board

NOTE: This type of "Bipolar" motor with connected-together center taps CAN be changed to "Unipolar" with the use of a different more complex driver board and modifying the motor as shown in THIS PFD

FILE

[RewireTheSmallStepperMotor.pdf](#)  
[Details](#) [Download](#) 171 KB

- LCD DISPLAYS
- SD CARDS
- Ultrasonic Sensors
- TemperatureSensors
- IR Remotes
- Stepper Motors
- RealTimeClock
- Keyboards/Buttons
- PWM Frequencies
- Analog Output
- TUTORIALS

BJK  
C

CONNECTION NOTES:

NOTE: *If your motor vibrates but does not turn or will only run in one direction*, it is probably connected with the wrong sequence.

The Arduino pin connections need to have 4 pins connected to Motor Driver In1, In2, In3, In4 and then the pins entered in the software in the sequence **1-3-2-4** for proper sequencing. Also, The + and - pins near "5-12V" need to be connected: - to Arduino Ground, + to Arduino +5 (for one motor test only) or (best) to a separate +5V 1A power supply.

Example: Connect Arduino Pins **8,9,10,11** to **In1,In2,In3,In4**

Then **software** is initialized in 1-3-2-4 sequence:

Stepper small\_stepper(STEPS, 8, 10, 9, 11); //Example Software Sketch below.

## STEP SEQUENCES:

The motor moves in response to the sequence in which the internal electromagnets are turned on. There are two possible sequences. In 4-step (Used by the Arduino Stepper Library) there are always 2 of the 4 magnet coils turned on, and only one coil changes each step.

The following refers to the letters A-B-C-D printed on the Stepper Driver Board which are controlled by the input pins 1-2-3-4. There are 4 LEDs next to the letters and they will follow the sequences. The test software sketch start out with a very slow sequence of the 4 step sequence. Push RESET on your Arduino to see that startup again.

4 Step Sequence : AB-BC-CD-DA (Usual application using Arduino STEPPER Library)

The 8-step sequence uses only 1 coil on, then 2, then 1... etc

8 Step : A - AB - B - BC - C - CD - D - DA - A (Can be done with other Libraries).

DRIVER LED LETTER	MOTOR CABLE#	MOTOR WIRE COLOR	step	step	step	step	step	step	step	step
4-STEP SEQUENCE			1		2		3		4	
8-STEP SEQUENCE			1	2	3	4	5	6	7	8
	5	red	+	+	+	+	+	+	+	+
D	4	orange	0	0	0	0	0	1	1	1
C	3	yellow	0	0	0	1	1	1	0	0
B	2	pink	0	1	1	1	0	0	0	0
A	1	blue	1	1	0	0	0	0	0	1

Here is test code for this Driver Board and Motor. Cut and Paste into a blank Arduino IDE page. NOTE: It uses the 4-step sequence shown above. So 2048 steps per output shaft revolution.

This uses the standard built-in Stepper library:

<http://arduino.cc/en/Reference/Stepper> . Other higher-performance libraries are available. See:

- [AccelStepper](#): Allows ramping up and down speeds, multiple motors, power down, more.
  - See [Example 2-motor sketch](#) below
- [Steve Bright's Stepper2](#) - Many functions, non-blocking modes, power control

See INSTALLING LIBRARIES [HERE](#):

### Test Sketch: 4-step sequence, Then 1/2 turn forward slow and back 1/2 turn fast

```
/* YourDuino.com Example Software Sketch
Small Stepper Motor and Driver V1.3 11/30/2013
http://arduino-direct.com/sunshop/index.php?l=product\_detail&p=126
Shows 4-step sequence, Then 1/2 turn and back different speeds
terry@yourduino.com */

/*-----( Import needed libraries )-----*/
#include <Stepper.h>

/*-----( Declare Constants, Pin Numbers )-----*/
//---( Number of steps per revolution of INTERNAL motor in 4-step mode )---
#define STEPS_PER_MOTOR_REVOLUTION 32

//---( Steps per OUTPUT SHAFT of gear reduction )---
#define STEPS_PER_OUTPUT_REVOLUTION 32 * 64 //2048

/*-----( Declare objects )-----*/
// create an instance of the stepper class, specifying
// the number of steps of the motor and the pins it's
// attached to

//The pin connections need to be 4 pins connected
// to Motor Driver In1, In2, In3, In4 and then the pins entered
// here in the sequence 1-3-2-4 for proper sequencing
Stepper small_stepper(STEPS_PER_MOTOR_REVOLUTION, 8, 10, 9, 11);

/*-----( Declare Variables )-----*/
int Steps2Take;

void setup() /*-----( SETUP: RUNS ONCE )-----*/
{
  // Nothing (Stepper Library sets pins as outputs)
}/*--(end setup )---*/

void loop() /*-----( LOOP: RUNS CONSTANTLY )-----*/
{
  small_stepper.setSpeed(1); // SLOWLY Show the 4 step sequence
```

```

Steps2Take = 4; // Rotate CW
small_stepper.step(Steps2Take);
delay(2000);

Steps2Take = STEPS_PER_OUTPUT_REVOLUTION / 2; // Rotate CW 1/2 turn
small_stepper.setSpeed(100);
small_stepper.step(Steps2Take);
delay(1000);

Steps2Take = - STEPS_PER_OUTPUT_REVOLUTION / 2; // Rotate CCW 1/2 turn
small_stepper.setSpeed(700); // 700 a good max speed??
small_stepper.step(Steps2Take);
delay(2000);

/* --(end main loop )-- */

/* ( THE END ) */

```

**Test Sketch: Rotate 1 turn in each direction, repeat. So this is a repeatable "back and forth" motion and is not dependent on the exactly precise gear ratio.**

```

/* YourDuino.com Example Software Sketch
   Small Stepper Motor and Driver V1.4 11/30/2013
   http://arduino-direct.com/sunshop/index.php?l=product_detail&p=126
   Steps one revolution of output shaft, then back
   terry@yourduino.com */

/*-----( Import needed libraries )-----*/
#include <Stepper.h>

/*-----( Declare Constants, Pin Numbers )-----*/
//---( Number of steps per revolution of INTERNAL motor in 4-step mode )---
#define STEPS_PER_MOTOR_REVOLUTION 32

//---( Steps per OUTPUT SHAFT of gear reduction )---
#define STEPS_PER_OUTPUT_REVOLUTION 32 * 64 //2048

/*-----( Declare objects )-----*/
// create an instance of the stepper class, specifying
// the number of steps of the motor and the pins it's
// attached to

//The pin connections need to be 4 pins connected
// to Motor Driver In1, In2, In3, In4 and then the pins entered
// here in the sequence 1-3-2-4 for proper sequencing
Stepper small_stepper(STEPS_PER_MOTOR_REVOLUTION, 8, 10, 9, 11);

/*-----( Declare Variables )-----*/
int Steps2Take;

void setup() /*-----( SETUP: RUNS ONCE )-----*/
{
  // Nothing (Stepper Library sets pins as outputs)
}/*--(end setup )---*/

void loop() /*-----( LOOP: RUNS CONSTANTLY )-----*/
{
  Steps2Take = STEPS_PER_OUTPUT_REVOLUTION ; // Rotate CW 1 turn
  small_stepper.setSpeed(500);

```

```

    small_stepper.step(Steps2Take);
    delay(1000);

    Steps2Take = - STEPS_PER_OUTPUT_REVOLUTION; // Rotate CCW 1 turn
    small_stepper.setSpeed(500); // 700 a good max speed??
    small_stepper.step(Steps2Take);
    delay(2000);

}/* --(end main loop )-- */

/* ( THE END ) */

```

**Test Sketch: Runs 2 Stepper motors in opposite directions, accelerates, decelerates, reverses.**

REQUIRES the Accelstepper Library [HERE](#)

See INSTALLING LIBRARIES [HERE](#):

```

/* YourDuinoStarter Example: 2 Stepper Motors
- WHAT IT DOES: Runs 2 28BYJ-48 stepper motors with AccelStepper Library
- Motors accelerate and decelerate simultaneously in opposite rotations
- SEE the comments after "/" on each line below
- Derived from example code by Mike McCauley
- modified by Celem for single stepper
- modified by lowres for two steppers
NOTE: This may not run 2 motors from USB.
      May need separate +5 Supply for motors
- CONNECTIONS: See Pin definitions below

- V1.01 11/30/2013
  Questions: terry@yourduino.com */

/*-----( Import needed libraries )-----*/
#include <AccelStepper.h>
/*-----( Declare Constants and Pin Numbers )-----*/
#define FULLSTEP 4
#define HALFSTEP 8
// motor pins
#define motorPin1 4 // Blue - 28BYJ48 pin 1
#define motorPin2 5 // Pink - 28BYJ48 pin 2
#define motorPin3 6 // Yellow - 28BYJ48 pin 3
#define motorPin4 7 // Orange - 28BYJ48 pin 4
// Red - 28BYJ48 pin 5 (VCC)

#define motorPin5 8 // Blue - 28BYJ48 pin 1
#define motorPin6 9 // Pink - 28BYJ48 pin 2
#define motorPin7 10 // Yellow - 28BYJ48 pin 3
#define motorPin8 11 // Orange - 28BYJ48 pin 4
// Red - 28BYJ48 pin 5 (VCC)

/*-----( Declare objects )-----*/
// NOTE: The sequence 1-3-2-4 is required for proper sequencing of 28BYJ48
AccelStepper stepper1(HALFSTEP, motorPin1, motorPin3, motorPin2, motorPin4);
AccelStepper stepper2(HALFSTEP, motorPin5, motorPin7, motorPin6, motorPin8);

/*-----( Declare Variables )-----*/
//none

void setup() //***** SETUP: RUNS ONCE *****/
{
    stepper1.setMaxSpeed(1000.0);
    stepper1.setAcceleration(50.0);

```

```

stepper1.setSpeed(200);
stepper1.moveTo(2048); // 1 revolution

stepper2.setMaxSpeed(1000.0);
stepper2.setAcceleration(50.0);
stepper2.setSpeed(200);
stepper2.moveTo(-2048); // 1 revolution

}//--(end setup )---

void loop() //***** LOOP: RUNS CONSTANTLY *****/
{
  //Change direction at the limits
  if (stepper1.distanceToGo() == 0)
    stepper1.moveTo(-stepper1.currentPosition());
  if (stepper2.distanceToGo() == 0)
    stepper2.moveTo(-stepper2.currentPosition());

  stepper1.run();
  stepper2.run();

}//--(end main loop )---

/*-----( Declare User-written Functions )-----*/
//none
/***** ( THE END ) *****/

```

ZZ

(<https://www.wikispaces.com/user/view/VincentPalagi>)



### Stepper sequence

VincentPalagi

(<https://www.wikispaces.com/user/view/VincentPalagi>)

Apr 8, 2016

I am having a problem understanding why the

#define

STEPS\_PER\_MOTOR\_REVOLUTION 32

defines the steps as 32 but the motor steps are 48

With the motor provided the 1 rotation program does not function properly either, it rotates further CW than CCW



(<https://www.wikispaces.com/user/view/dcromley>)



[Help](#) · [About](#) · [Blog](#) · [Pricing](#) · [Privacy](#) · [Terms](#) · **[Support](#)**

Contributions to <https://arduino-info.wikispaces.com/> are licensed under a Creative Commons Attribution Non-Commercial 3.0 License. 

Portions not contributed by visitors are Copyright 2017 Tangient LLC

TES: The largest network of teachers in the world