

Microstepping Tutorial

Rob Steves

Introduction

A bipolar stepper motor has two windings. The current through each winding is varied in order to rotate the stepper motor. When considering stepper motor drive techniques, a “phase diagram” is a useful visualization tool. The current through one winding I_a is plotted against the current through the other winding I_b . Modes of operation such as full-stepping, half-stepping, microstepping, and operation at different current limits can be easily visualized on such a diagram. In addition, it is possible to visualize changes in both power consumption and torque as a function of angular position. Simple stepper motor controllers are only capable of driving a winding with full-positive current, no current, or full-negative current. Given these available outputs, it is only possible

to implement full-stepping, half-stepping, or wave-stepping.

The arrows in each phase diagram are called “phasors.” The angle theta that the phasor moves from one position to the next is the step- or microstep-angle. On a phase diagram, 90° corresponds to one full-step and 360° corresponds to a “full-step sequence.” A full-step sequence is a sequence of steps or microsteps which, when repeated, will produce continuous rotation of the motor. Assuming adequate torque, any continuous path that traverses the four quadrants of the phase diagram with at least one point-per-quadrant will suffice to rotate the stepper motor. If the controller is designed with the capability to control the magnitude of the current in each winding, then microstepping can be implemented. The phase diagrams (Fig. 2) all show different implementa-

tions of “divide-by-four” microstepping. Note that it is the phasor angle (not its length) that determines the microstep position. The phasor length affects power consumption and available torque, as we will see later.

While it is convenient to think of the I_a and I_b axes as representing full-step positions, it should be recognized that this is an arbitrary choice and any four positions in the phase diagram that are 90° apart from each other could be considered full-step positions. However, for the sake of simplicity, let us consider the positive I_a axis to represent $\theta=0^\circ$. As theta increases, the phasor moves counter-clockwise from this position. At $\theta=90^\circ$ the phasor lies along the positive I_b axis—one full-step from its starting position. Any angle theta between 0 and 90° represents a possible microstep position (a

FULL-STEPPING. In full-stepping operation, the current required in each winding is either $-I_{max}$ or $+I_{max}$. A step sequence of four full-steps makes up one complete step cycle. Note that these full-step positions are the same as the odd numbered positions from the half-stepping sequence.

HALF-STEPPING. In a half-stepping operation, the current required in each winding is either $-I_{max}$, 0, or $+I_{max}$. A step sequence of eight half-steps makes up one complete step cycle.

WAVE-STEPPING. Wave-stepping is another method of full-stepping, but with reduced power requirements (and corresponding torque output), since only one winding is powered at a time. The current required in each winding is either $-I_{max}$, 0 or $+I_{max}$. A step sequence of four full-steps makes up one complete step cycle. Note that these full-step positions are the same as the even numbered positions from the half-stepping sequence.

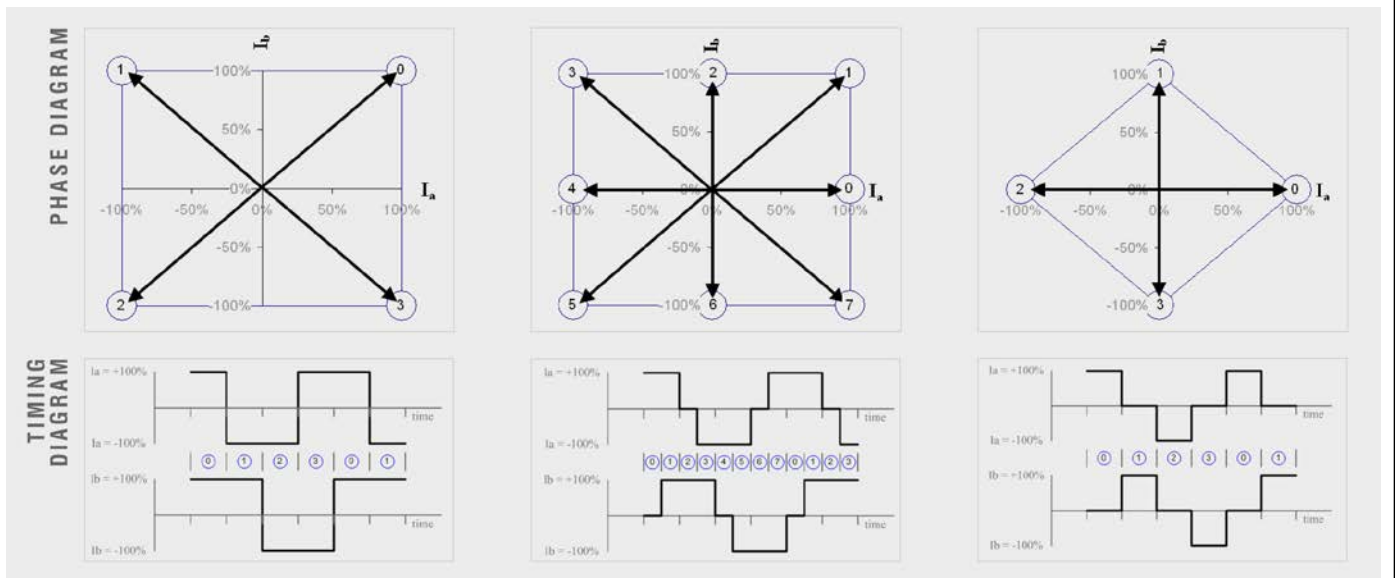


Figure 1

Reprinted courtesy of Zaber Technologies (www.zaber.com).

MICROSTEPPING: square path. This method of microstepping provides the highest peak torque if you are limited by available supply voltage.

MICROSTEPPING: circular path. This method is also referred to as sine-cosine microstepping and is usually what people are referring to when they talk about microstepping, although it is in fact only one method.

MICROSTEPPING: arbitrary path. There would be little reason to use a method such as this. It is presented only to illustrate the possibilities. Although it looks very strange compared to the other two methods, in theory it will produce the same angular rotation of an ideal motor. Only the available thrust would differ.

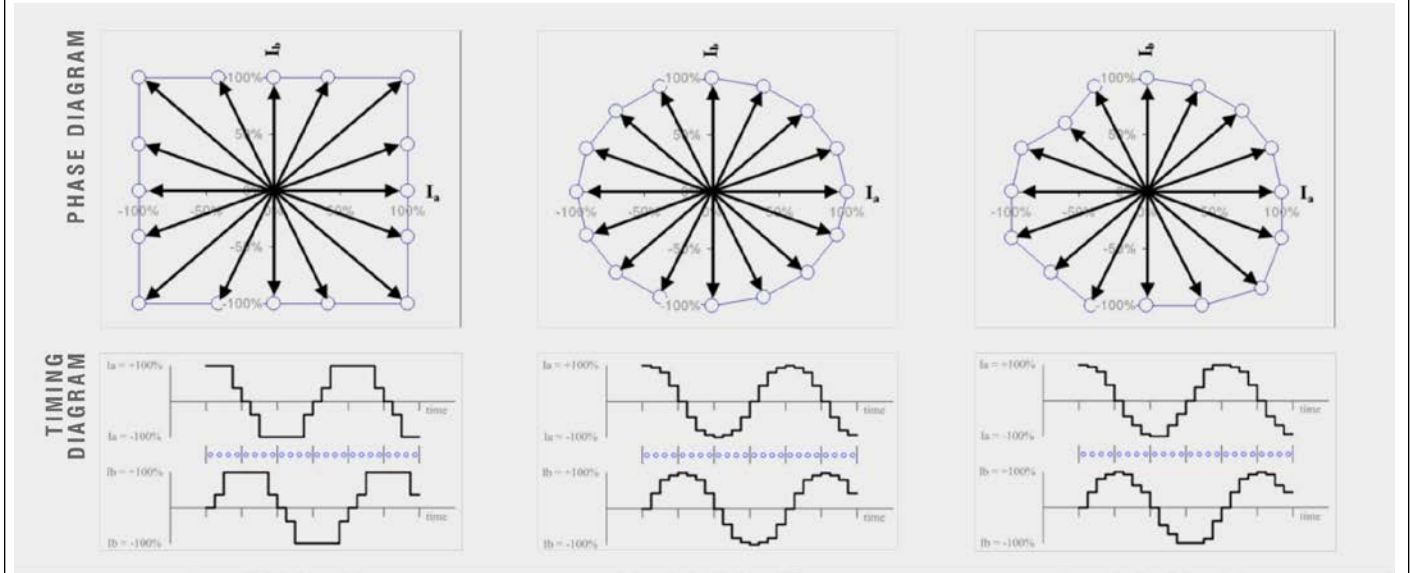


Figure 2

position between full-step positions). If you wish to implement “divide-by-10” microstepping, then you must generate values of I_a and I_b that correspond to values of theta equal to 0, 9, 18, 27 ... 81, 90°, etc.

$$\theta = \tan^{-1}(I_b/I_a) \quad (1)$$

There are many values of I_a and I_b that could be chosen to produce the same phasor angle theta. The choice of phasor length is typically decided based on the motor power rating and the application’s torque requirements. Power and torque are both related to phasor length. The power draw at any given angular position is given by the formula:

$$\text{power} = I_a^2 R + I_b^2 R \quad (2)$$

where:

R is the winding resistance (both windings should have the same resistance)

The length of the phasor is given by the formula:

$$\text{phasor_length} = \sqrt{I_a^2 + I_b^2} = \sqrt{\text{Power}/R} \quad (3)$$

Thus the phasor length gives an indication of the power draw at each microstep angle. Torque is directly proportional to current (assuming magnetic saturation is not reached). As such, the available torque is directly proportional

to phasor length, and the phase diagram gives an indication of how torque may vary with microstep position.

(Aside: note that phasor length is proportional to torque and to $\sqrt{\text{power}}$. Therefore torque is proportional to $\sqrt{\text{power}}$. In other words, a 2x increase in torque requires a 4x increase in power—assuming magnetic saturation is not reached.)

A phasor of constant length is typically used for smoothest operation (minimum torque ripple) and constant power output. This results in a circular path around the phase diagram, the phasor length being the radius of the circle. This technique is referred to as “sine cosine microstepping” because the target values for I_b and I_a are proportional to $\sin(\theta)$ and $\cos(\theta)$, respectively. In practice, the term microstepping usually refers to sine cosine microstepping; but in theory, sine cosine microstepping is only one method of microstepping. As mentioned above, a functioning microstepping algorithm may be designed around any arbitrary path that traverses the four quadrants of the phase diagram and has at least one point per quadrant.

The maximum continuous power output of a motor is usually specified

by the manufacturer. This places an upper limit on phasor length, $\sqrt{I_a^2 + I_b^2}$. The effective limit that this places on I_a and I_b will depend on the geometry of the path chosen around the phase diagram.

One must also consider the limitations of the power supply being used to drive the motor. In voltage-controlled products, the maximum values of I_a and I_b may be limited by the power supply voltage ($I = V/R$). In this case, using a square phase profile is a way to achieve higher torque without requiring a higher voltage power supply.

Zaber devices are driven in a circular phase path to achieve smooth motion and constant power output. The current limit is user-configurable and specifies the value of I_a at $\theta = 0$. In some Zaber devices (T-series), square-phase mode is also available. Square-phase mode offers about 40 percent higher torque for the same current limit. However, there is a price to pay—i.e., torque ripple. You can see that as you move around the phase diagram, along a square path, the torque (proportional to the length of the phasor) will be constantly increasing and decreasing. This results in less smoothness of operation and less microstepping accuracy. Generally, it is easier to achieve higher torque simply by increasing the current limit. However, at times your maximum current limit is restricted by your supply voltage or the capabilities of your controller. In this case, reduced smoothness and accuracy may be an acceptable compromise for additional torque.

Sources of Error in Microstepping Systems

Stepper motor control systems are usually open-loop. That is, the controller does not have position feedback and therefore is not aware of the “actual” position of the motor. Therefore, it is important to be aware of possible sources of error that will result in the actual position being different from the calculated position.

Quantization error. In any digital controller, it is impossible to achieve infinitely variable I_a and/or I_b —only discrete or “quantized” values are possible. The number of discrete values depends on the resolution achievable by the controller. For example, if the maximum current output of the controller is 1A, and the controller has a resolution of 0.1A, then there are 10 possible current values for I_a and/or I_b , not including 0. The number of discrete values possible determines how close mathematically the phasor can be set to a particular length and microstep angle. The error between the desired phasor angle and the actual phasor angle achieved is the quantization error.

A maximum quantization error equivalent to 0.5 microsteps is a typical design requirement in any microstepping control algorithm. Note that by adjusting the phasor end-point to a nearby I_a/I_b point, rather than sticking to a strictly circular or square profile, can often reduce the quantization error, but may add some torque ripple. Thus, the current resolution you require for I_a and I_b will be determined by the number of microsteps-per-step you want to achieve, the quantization error you can

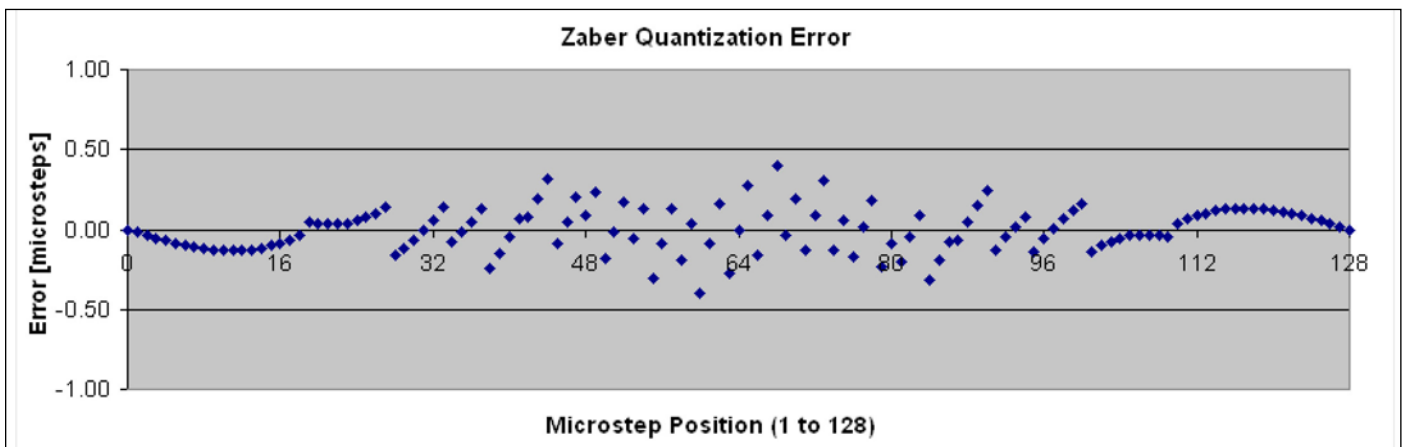


Figure 3 — Zaber quantization error.

tolerate, and the torque ripple you can tolerate.

In Zaber's stepper motor control algorithm, the design requirements were 128 microsteps-per-step, with a quantization error less than 0.5 microsteps and a torque ripple less than 2.5 percent. Determining how many discrete current values are required for I_a and I_b is a task best left to a spreadsheet application, such as *Excel*. Even then, it requires a certain degree of trial and error. As it turns out, 80 discrete current settings (between 0 and the running current) are required to achieve 128 microsteps-per-step with a quantization error less than 0.5 microsteps, and a torque ripple less than 2.5 percent. The resulting quantization error at each microstep position is plotted (Fig. 3).

Detent error. Detent torque is the maximum torque that can be applied to a non-energized stepper motor without causing continuous rotation. If you plotted torque-vs.-shaft-angle as you slowly rotate the stepper motor—with no current in either winding—you then would find that the torque is approximately sinusoidal with shaft angle. The detent torque is just the amplitude of the sine curve. In an ideal motor the torque curve would be perfectly sinusoidal. What is commonly referred to as "detent error" isn't due to the existence of the detent torque per se, but due to the non-sinusoidal component of the detent torque. The shape of the torque curve is affected by motor pole geometry. In that sense, detent error is really pole geometry error. Because different motor manufacturers use different pole geometries, this error can vary from one manufacturer to another, as well as from one motor to another.

Motor pole placement error. Motor pole placement error results in a varying step size. There is typically an error that repeats every four steps (one complete step cycle), as well as an error that repeats every full revolution. This has an obvious effect on microstepping. The microstep size within large steps will be proportionally larger than the microstep size in small steps. Pole placement error in a typical motor is less than 0.5 steps of cumulative error over half a revolution of the motor. Given that a typical motor has 200

steps-per-revolution, that translates to an error in step size of roughly ± 0.5 percent. It is possible to eliminate pole placement error in any application simply by moving in increments of one full revolution of the motor. If that is not possible, then some error can be eliminated by moving in increments of four steps. However, moving in increments of four steps or full revolutions is clearly not microstepping. Therefore, all microstepping applications invariably suffer from some pole placement error.

Leadscrew pitch error. Many motorized systems convert rotary motion to linear motion via leadscrew; stepper motor applications are no exception. In these types of systems, any error in the leadscrew pitch will contribute to total system error.

Stiction and backlash error. In microstepping systems, mechanical stiction and backlash are frequently much larger than the microstep resolution. There are many systems on the market capable of microstepping at 256 microsteps-per-step, but there is little point to this if mechanical stiction in the system will be on the order of five to ten microsteps at that microstep resolution.

Conclusion

This discussion has centered on the challenges of designing a microstepping system, but there are also challenges when implementing any system. If the load on a stepper motor exceeds its maximum torque, then the motor poles will not follow the changing magnetic field and the motor stalls. To avoid this type of failure, microstepping systems must either keep the load below the maximum torque, or include position sensors to detect and compensate for stalls.

For more information:

Zaber Technologies Inc.
1777 West 75th Ave
Vancouver, British Columbia
Canada, V6P 6P2
Phone: (888) 276-8033 (Toll Free Canada/USA)
(604) 569-3780 (Direct)
Fax: (604) 648-8033
contact@zaber.com

Rob Steves, B.A.Sc., M. Eng. is an electromechanical design engineer and president of Zaber Technologies Inc., dedicated to the design and manufacture of stepper motor-based precision linear actuators, linear slides, and other motion control products used for optics, industrial automation, biomedical, and many other applications.