Measurement and Identification of Brush, Brushless, and dc Stepping Motors

Stuart Schweid
Xerox Corporation

Robert Lofthus
Xerox Corporation

John McInroy
University of Wyoming

101.1 Introduction

There are many systems, such as robots, whose ability to move themselves or other objects is their primary purpose. There is a myriad of other systems, such as xerographic printers, where the motion is not the desired outcome but is required in the performance of the mainline objective. All of these systems require one or more "prime movers," which directly or indirectly create all motion in the system.

All of these applications have differing requirements for both the type and precision of the motion produced. There are applications, such as children's toys, that can perform well with imprecise motion requirements. Conversely, there are applications, such as printing, that require very precise motion control.

Depending on the type of motion required, a prime mover can be chosen from one of several candidates. The most ubiquitous motor types are dc brush type motors, dc brushless motors, and hybrid stepping motors. The system, which includes the prime mover, can be controlled simply as a function of time (i.e., open loop) or as a function of both time and system state (i.e., closed loop).

A closed-loop system requires some mechanism of measuring the "state" of the system. For motion systems, this is typically one or more sensors that measure position and/or velocity. In addition, for many of the control techniques, specifically, linear controllers, it is also advantageous to have a model of the system being controlled. The model can be used in conjunction with a plethora of design techniques to improve the dynamic response of the system significantly.

This chapter will describe the major modes of operations for the prime movers listed and will detail a method for measuring the open-loop response of a candidate motion system.

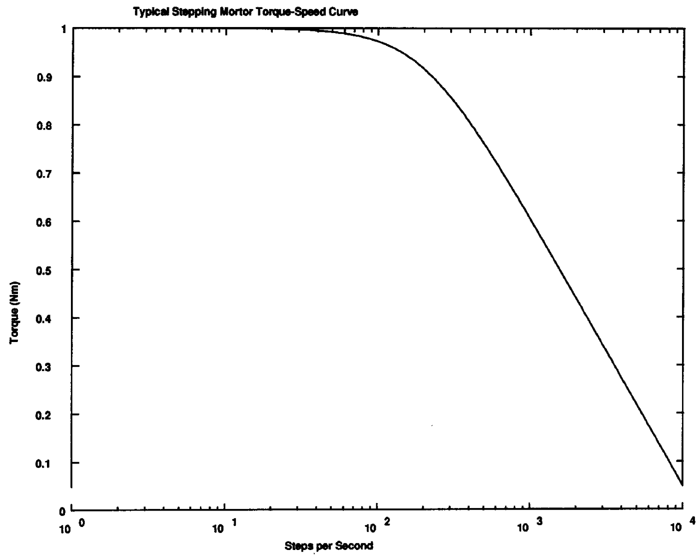

FIGURE 101.1 Typical torque–speed curve.

101.2 Hybrid Stepping Motors

Hybrid stepping motors are useful devices that provide fairly accurate positioning in the open loop: their most common mode of operation. They are successful in open-loop operation because they will remain within a commanded position increment or step as long as the motor has sufficient torque to resist any external torque applied by the system.

Since the motor operates in the open loop, it is chosen *a priori* to ensure it provides the torque necessary to remain within step. The maximum torque needed to operate the system is a function of the inertia of the system, the acceleration profile of the system, the external torque on the system, etc. Once the maximum torque required at every speed is determined, a motor can be chosen from its torque–speed curve. The torque–speed curve describes the maximum torque that can be tolerated at each speed without failing to follow a commanded step (referred to as a "losing step"). A motor whose torque–speed curve exceeds the torque requirements of the system at all speeds will be sufficient for the application. Figure 101.1 shows a typical torque–speed curve.

Hybrid stepping motors have a plurality of coils (almost always two) that, when commutated in a predefined sequence, cause the motor to turn. When the motor is operated in a mode referred to as "full stepping," the current command to each winding is constant in magnitude, but varying in polarity (positive or negative). Note that the definition of which direction of current flow is "positive" is purely arbitrary. In this mode, the motor has only four separate energizations of the winding to complete a full revolution, although clever mechanical arrangements allow the number of steps per revolution to be greatly increased: typically 200 but as large as 800.

With two distinct windings (referred to as A and B), the four possible arrangements of winding energizations are A+B+, A+B–, A–B–, A–B+. If these combinations of winding currents are commanded to the motor in the sequence listed in the previous sentence, the motor will turn.

Chopping Current Amplifiers

As previously stated, the motion of the stepper requires a current of a particular amplitude to be applied. For economic reasons, most systems have a voltage power supply available (V_{ss}) — not a current source. A "current chopper" is a lost cost closed-loop switching amplifier that regulates the current to the motor winding. It takes advantage of the fact that the motor winding has a significant inductance. The two motor windings have dynamic equations [1]

$$dI_A/dt = \left(V_A - I_A R_A + K_b \omega \sin\theta\right)/L_A$$
$$dI_B/dt = \left(V_B - I_B R_B + K_b \omega \cos\theta\right)/L_B \qquad (101.1)$$

where I is the current winding (A), R is the current resistance (Ω), L is the current inductance (H), V is the applied voltage to the winding (V), K_b is the torque constant of the motor (Nm/A), ω is the motor velocity (rad/s), and θ is the motor position (electrical rad).

Every t_s s (where t_s is very small, i.e., 10 µs) the current chopper applies either V_{ss} or $-V_{ss}$ to the motor winding, depending on whether the desired current (I_M) is greater than or less than the measured (i.e., actual) winding current:

$$\text{if} \quad \left(I_A > I_M\right), \quad V_A = -V_{ss} \quad \text{else} \quad V_A = V_{ss} \qquad (101.2)$$

$$\text{if} \quad \left(I_B > I_M\right), \quad V_B = -V_{ss} \quad \text{else} \quad V_B = V_{ss}$$

The usual requirement is that (V_{ss} t_s/L) is small in comparison with I_M, resulting in a the chopping range that is small when compared with the desired current. Furthermore, the mechanical system that includes the motor is typically low pass in nature, so it will have minimal response to the small-amplitude, very high frequency components present in the motor winding current.

Microstepping

Another common open-loop mode for stepper operation is microstepping. In microstepping, the commanded reference positions can be between the "full-step" positions of the motor. The commanded currents for the reference position θ_{ref} are

$$I_A = I_M \cos\left(\theta_{ref}\right)$$
$$I_B = I_M \sin\left(\theta_{ref}\right) \qquad (101.3)$$

where θ_{ref} is the desired electrical position of the motor (in four full steps, θ_{ref} traverses one full electrical revolution). For constant velocity applications,

$$\theta_{ref} = N\omega_d t \qquad (101.4)$$

where N is the number of electrical revolutions per mechanical revolution (one quarter the number of steps per revolution) of the motor and ω_d is the desired velocity (rad/s).

Closed-Loop Control

In addition to the open-loop uses of hybrid stepping motors, several researchers have recently invented methods of operating these motors in a closed-loop fashion. The principal difficulty arises because hybrid stepping motors are nonlinear devices, so the plethora of design techniques available in linear control theory cannot be directly applied. Hamman and Balas [1] use a fourth-order model to linearize the system about each step. This is useful in altering the undamped response of the stepper motor to a commanded change in position. Schweid [2] suggests a scheme that allows good transient behavior during constant velocity applications, but uses the complete fourth-order model. In systems that use current choppers, a second-order model is sufficient. The following control law can then be applied [3-5]:

$$
\begin{aligned}
I_A &= I_M \cos(\theta_{ref}) - I_C \sin\theta \\
I_B &= I_M \sin(\theta_{ref}) + I_C \sin\theta
\end{aligned}
\tag{101.5}
$$

The currents of Equation 101.6 linearize the model of the stepper motor, allowing common control techniques such as PID control to be applied [6]. The value I_M is used as an open-loop component to microstep the motor, while I_C is a feedback term that permits an improved dynamic response.

Position Measurement

The implementation of closed-loop control requires a method of measuring the position of the motor. Furthermore, there is the added constraint that this position measurement must be aligned with the motor construction and not have a positional offset with respect to it. This is because the control scheme of Equation 101.5 requires a command current I_C be multiplied by the sine and cosine of the motor position. One way to implement this is to have an incremental encoder with an index pulse attached to the motor. A calibration can then determine the relative position of the index pulse to the motor shaft and use this relationship in all future positional measurement.

Another scheme uses the back emf of the two coil windings to estimate the position. Some motors are constructed with additional sensing coils that allow easy measurement of these values. From Equation 101.2, the back emf of both coils can be found using

$$
\begin{aligned}
V_{emfA} &= -K_b \omega \sin\theta = V_A - I_A R_A - L_A dI_a / dt \\
V_{emfB} &= -K_b \omega \cos\theta = V_B - I_B R_B - L_B dI_B / dt
\end{aligned}
\tag{101.6}
$$

and the position could be found using

$$\theta = \mathrm{atan}\left(-V_{emfA} / V_{emfB}\right) \tag{101.7}$$

In most motors, however, V_{emfA} and V_{emfB} are not directly measurable. However, it is possible to obtain an excellent estimate of position in constant-velocity system by approximating the back emf signals through low-pass-filtered measurements of the coil voltages. One method developed at Xerox that has a U.S. patent (U.S. 5378975) is described here. Taking the Laplace transform of Equation 101.6 yields

$$
\begin{aligned}
V_{emfA}(s) &= V_A(s) - (R_A + sL_A) I_A(s) \\
V_{emfB}(s) &= V_B(s) - (R_B + sL_B) I_B(s)
\end{aligned}
\tag{101.8}
$$

Filtering both sides with low-pass analog filters matched to the dynamics of the coil produces:

$$
\begin{aligned}
V_{filtA}(s) &= V_A(s) \Big/ \left(s\left(L_A / R_A\right) + 1\right) - R_A I_A, \\
V_{filtB}(s) &= V_B(s) \Big/ \left(s\left(L_B / R_B\right) + 1\right) - R_B I_B
\end{aligned}
\tag{101.9}
$$

The quantities V_{filtA} and V_{filtB} are easily obtained by low-pass filtering the coil voltage with a simple *RC* circuit and subtracting the *IR* term. The subtraction is easily implemented because the current is a known input (in fact it is commanded), and the resistance is easily measured.

Assuming that the coils are matched: $L = L_A = L_B$, $R = R_A = R_B$ and the motor is moving at constant velocity, $\theta = N\omega_d t$, the back EMF voltages are sinusoidal. The low-pass-filtered versions of these voltages are given by

$$V_{filtA}(s) = \left|H\left(N\omega_d\right)\right| K_B \sin(\theta + \phi)$$
$$V_{filtB}(s) = \left|H\left(N\omega_d\right)\right| K_B \cos(\theta + \phi) \qquad (101.10)$$

where $|H(N\omega_d)|$ is the magnitude response of the low-pass filter at the input frequency, $N\omega_d$, and $\phi = -\text{atan}(N\omega_d\, L/R)$ is the phase shift due to the filtering. Dividing the two Equations 101.10 and taking the inverse tangent yields

$$\theta = \text{atan}\left(-V_{filtA}(t)/V_{filtB}(t)\right) + \text{atan}\left(N\omega_d L/R\right) \qquad (101.11)$$

The full derivation and analysis are found in Reference 7. In addition, Reference 7 includes a Kalman filtering technique to improve the velocity measurement estimate from the position estimate. This is needed because the derivative or difference of the position measurement is highly sensitive to noise.

101.3 dc Brush and Brushless Motors

The dc servomotor is the oldest and probably the most commonly used actuator for servo systems. The name arises because a constant (or dc) voltage applied to the motor will produce motor movement. Over a linear range these motors exhibit a torque that is proportional to the current flowing in the winding. The chief disadvantage of the brush dc motor is the brush, which performs the mechanical commutation. These brushes are the first part of the motor to fail, severely limiting the reliability of dc brush motors when compared with other actuators.

Another motor that operates similarly to a dc brush motor is the dc brushless motor (a dc voltage will cause it to rotate). Its method of commutation, however, is much different. Whereas the dc motor accomplishes switching mechanically with brushes, the dc brushless motor contains Hall effect sensors that electronically control the switching. As a result, dc brushless motors have much greater life and reliability.

Pulse Width–Modulated Power Amplifier

In many systems where they operate, dc motors are required to have a varying voltage at their input in order to create the position or velocity profiles desired. Many of these systems, however, have only a single power supply V_{ss}. Rather than using analog power amplifiers that are both expensive and inefficient, a pulse width–modulated (PWM) amplifier is used. A pulse width power amplifier can deliver only V_{ss} or $-V_{ss}$ (or zero in some designs) V to the motor. Different voltages are delivered to the motor by varying the relative percentage that V_{ss} and $-V_{ss}$ are applied to the motor at a high fixed frequency. The mean or average voltage to the motor is a linear function of this relative percentage or duty cycle. The motor also receives the higher harmonics of the switched input that are integer multiples of the base frequency of the square wave. Most motors, however, have at least two poles (an electric and a mechanical) which are significantly slower than the switching frequency and thus greatly attenuate any possible effect they might have on motion. Figure 101.2 shows a PWM amplifier set to deliver 12 V to the motor.

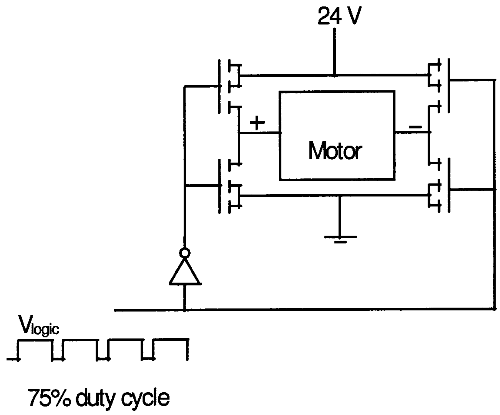

FIGURE 101.2 PWM amplifier set to deliver 12 V to the motor.

Measuring the System Dynamics

In systems with moderate or stringent motion requirements, the dc brush and brushless motors are operated in a closed-loop fashion. In order to perform in closed loop, some measure of either position or velocity is needed. A tachometer is commonly used to measure rotational speed. It outputs a voltage that is proportional to the speed of a rotating member. If a position measurement is required, an incremental encoder can be used. The incremental encoder produces a digital pulse at equal points in position. For example, a 1000 line/revolution encoder would produce a digital wave with 1000 rising (and falling) edges during one revolution.

Once the full system (including feedback sensors) is determined, a feedback control algorithm can be developed. As part of the design of many compensation algorithms, especially linear compensators such as state feedback or PID, one of the most important tasks is to have a good model of the dynamics of the system.

There are several ways to generate the model. One of the most common methods is to create the state equations of the system from the physical equations. This requires knowledge of all the parameters of the system. Sometimes these parameters are hard to measure and are not provided by the manufacturer. One example of this might be the torsional constant of a hard rubber roll in the system.

Even if all of the parameters are known, it is still desirable to have a method that measures the response of the system. This provides a way to either validate the model or to inform the designer that the model is not an accurate representation of the system.

A method is presented that permits the collection of input/output data of a system. This data can then be used to fit a frequency response curve or a transfer function model of the system. Signal analyzers, such as the HP3562A, have the ability to perform such curve fitting accurately. In addition, if the data are collected in a discrete fashion, there are commercially available software tools, such as MATLAB, that can provide model estimation in either a parametric (e.g., state space/transfer function) or nonparametric (e.g., frequency response) form. Although this procedure is described for motor systems, it is easily implemented in most linear systems.

Assume a typical closed-loop system as shown in Fig. 101.3. There are several obstacles that make it difficult to obtain the open- or closed-loop transfer function of this system. These obstacles include the inability to measure some of the outputs and the inability to control/modify certain inputs.

To acquire the frequency response of a system it is useful to perturb the system with a known input and measure the elicited response. A controllable input — one that can be externally manipulated — is required to accomplish this task. The only controllable inputs of the system in Fig. 101.3 are the torque acting upon the motor shaft and the reference command.

Changing the external torque would require mechanically attaching a calibrated brake or another motor. This is not only cumbersome, but it also can alter the system dynamics, as it introduces an inertia

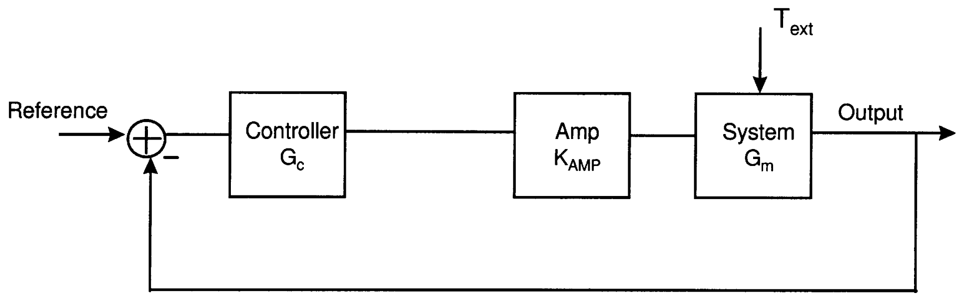

FIGURE 101.3 Typical closed-loop system.

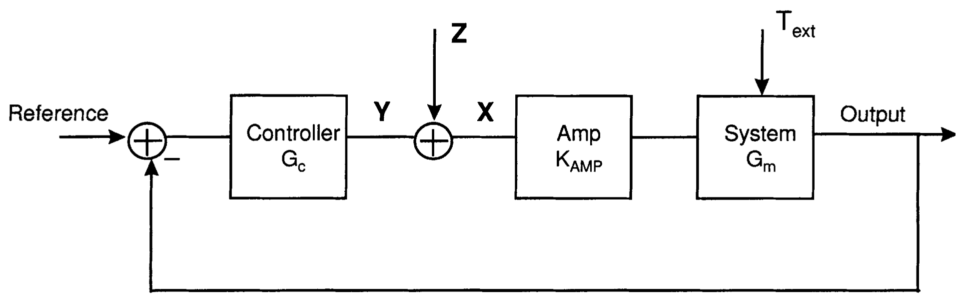

FIGURE 101.4

to the true system under measurement. Within many systems, the reference signal is a digital square wave (since the feedback element is an incremental encoder and produces a similar wave), and therefore requires a voltage-to-frequency converter if the reference is to be directly manipulated. This converter may include dynamics that can corrupt the measured data.

In addition to a controllable input, a measurable output must be available. In many servo systems, the only measurable outputs are the motor armature voltage and the position/velocity. The armature voltage may be a PWM signal that needs to be filtered to create an analog voltage. In the case of an incremental encoder, the output consists of a digital square wave generated by an incremental encoder: a frequency-to-voltage converter is required to produce an analog output. This converter may include dynamics that can corrupt the measured data.

Consider the system described in Fig. 101.4. The summing junction placed in the system has the advantage that can be added at any point that is convenient for measurement purposes — it is no longer limited to unaugmented system inputs or outputs.

In the case of analog controllers, the inputs and outputs of the summing junction are analog voltages, making them easily controlled and measured. The summing node can be easily implemented using a simple op-amp adder circuit. This introduces no appreciable dynamics into the original system and will not, therefore, skew the measured data.

When the system response is to be measured, the time-varying contributions of the other system inputs: the velocity reference and external torque, T_d, acting upon the system must be eliminated. The designer must set the velocity reference as a constant value. There is usually little control of the external torque acting on the system, and this will introduce noise into the measurement technique. To ensure a large signal-to-noise ratio (SNR), the alternative reference input (signal $z(t)$) should be made as large as possible.

The alternative reference input, $z(t)$, can stimulate the system with a swept sine input, a random input, or any other desired signal. During the stimulation, a time record is collected of the other nodes of the summing junction, $y(t)$ and $x(t)$. The transfer function between Y and X is

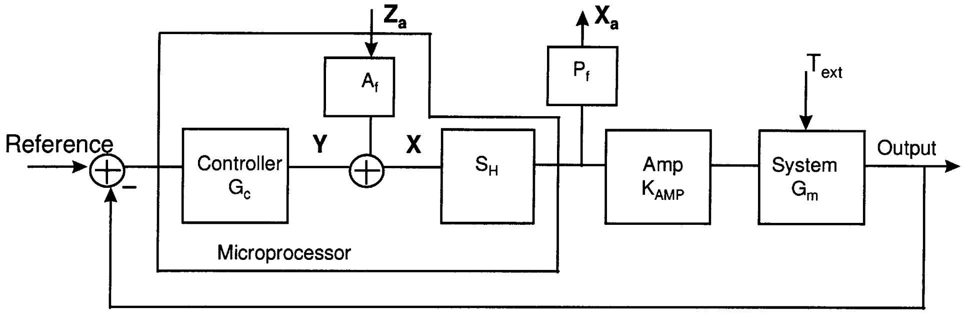

FIGURE 101.5

$$\mathbf{Y}(s)/\mathbf{X}(s) = -G_c(s)K_{amp}(s)G_m(s), \quad -\mathbf{Y}(s)/\mathbf{X}(s) = G_c(s)K_{amp}(s)G_m(s) \qquad (101.12)$$

The second term is, by definition, the open-loop transfer function of the system. As before, MATLAB or a signal analyzer can provide either a parameteric or nonparametric fit of this transfer function from the data record.

Note that this technique requires the system be functioning in a closed-loop mode during data collection. Typically, G_c is set as a proportional controller at a value that will ensure stability of the overall system. For a good data collection, the proportional gain should be made as large as possible.

The case of a digital or microprocessor-controlled system can be handled similarly, with some slight modification. Consider the system of Fig. 101.5, where a microprocessor acts as the system controller. In this instance, the summing junction can be implemented internal to the microprocessor by executing an add instruction. The problem here becomes getting the data into and out of the microprocessor. Getting the data into the microprocessor requires the addition of an analog-to-digital converter (ADC). In the block diagram, this ADC process is represented by the block A_f. A_f allows representation of any dynamics inherent in the ADC design (especially gain). The output of the microprocessor may be a digital PWM signal. This signal may need to be low-pass-filtered to create an analog voltage for the signal analyzer taking the data. The low-pass filter is represented by P_f in the Fig. 101.5. It is typically a simple *RC* circuit whose cutoff frequency is much lower than the PWM signal frequency.

Given the condition that ω_{ref} is static and external torque, T_d, is minimal, the transfer function between $\mathbf{Z_a}(s)$ and $\mathbf{X_a}(s)$ in Fig. 101.5 is

$$\mathbf{Z_a}(s)/\mathbf{X_a}(s) = \left(1 + G_c(s)S_H(s)K_{amp}(s)G_m(s)\right) \Big/ \left(A_f(s)S_H(s)P_f(s)\right) \equiv \mathbf{O}(s) \qquad (101.13)$$

when $G_c = 0$, $\mathbf{Z_a}(s)/\mathbf{X_a}(s) = 1/(A_f(s)S_H(s)P_f(s)) \equiv \mathbf{C}(s)$, $(\mathbf{O}(s) - \mathbf{C}(s))/\mathbf{C}(s) = G_c(s)S_H(s)K_{amp}(s)G_m(s)$, this is the system open-loop transfer function.

A more direct approach is possible if a logic analyzer is not used but a software-based system identification package such as MATLAB is used to fit the data. The stimulus, z, can either be stored as a set of data in a lookup table in the microprocessor memory or can be generated in real time using any random number generator algorithm or other algorithm. The data Y and X at the internal summing junction can be captured via a logic analyzer or emulator if the values of $y(k)$ (the value of y at the kth sample period) and $x(k)$ are written to an external bus every sample period. Under this configuration neither a DAC nor ADC is necessary, nor is any mathematical combination of two different transfer functions required to produce the desired open-loop transfer function. The transfer function between $\mathbf{Y}(z)$ and $\mathbf{X}(z)$ is

$$\begin{aligned} \mathbf{Y(z)}/\mathbf{X(z)} &= -G_c(z)S_H(z)K_{amp}(z)G_m(z) \\ -\mathbf{Y(z)}/\mathbf{X(z)} &= G_c(z)S_H(z)K_{amp}(z)G_m(z) \end{aligned} \qquad (101.14)$$

Again, this is the open-loop transfer function of the system, acquired using only minimal additional software and a logic analyzer. As before, the input/output set of data can be sent to any system identification algorithm to fit the transfer function.

References

1. E. Hamman, "Closed-loop control of a DC stepping motor using state space techniques," Master's thesis, Rensselaer Polytechnic Institute, Troy, NY 1983.
2. S. Schweid, "Velocity control of DC stepping motors utilizing state space theory, quasilinearization and reduced order estimation/control," Masters thesis, Rensselaer Polytechnic Institute, Troy, NY 1985.
3. J. Tal, L. Antignini, P. Gandel, and N. Veignat, "Damping a two-phase step motor with velocity coils," *Incremental Motion Control Systems and Devices Symposium*, Champaign, IL 1985, 305–309.
4. D. Reignier, "Very accurate positioning system for high speed, high acceleration motion control," *Power Conversion and Intelligent Motion*, 13, 52–62, June 1987.
5. D. Reignier, "Damping circuit and rotor encoder cut disc magnet step motor overshoot, settling time and resonances," *Power Conversion and Intelligent Motion*, 14, 64–69, April 1988.
6. S. Schweid, J. McInroy, and R. Lofthus, "Closed loop low-velocity regulation of hybrid stepping motors amidst torque disturbances," *IEEE Trans. Ind. Electron.*, 42, 316–324, 1995.
7. R. Lofthus, S. Schweid, J. McInroy, and Y. Ota, "Processing back EMF signals of hybrid step motors," *Control Eng. Practice*, 3(10), 1–10, 1995.