Introduction to BLDC Motor Control
Using Freescale MCU

Tom Wang
Segment Biz. Dev. Manager
Avnet Electronics Marketing Asia
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
Structure of a single-phase 2 pole-pair BLDC motor

- Based on attraction and repulsion of magnetic poles.
- Rotating magnetic field of stator rotates rotor
- Rotation of magnetic field must be in phase with rotation of rotor.
- Rotation of magnetic field is achieved by changing direction of current through the stator coil.
- Speed Control achieved by controlling the average current flowing into stator coil.
- Hall Sensor used to detect rotor position.
Operation of a single-phase 1 pole-pair BLDC motor
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
Motor Mechanical Model

Mechanical model:

Mechanical Drive Torque generated $T$ is expressed as follows:

$$T = K_T \cdot i \cdot \sin(N \cdot \theta + C) = J \cdot d\omega/dt + L \cdot \omega + T_{ext}$$

where:

- $K_T$ = torque constant
- $i$ = current through coil
- $N$ = Number of pole pairs
- $\theta$ = phase angle of rotor
- $C$ = phase offset due to motor phases
- $J$ = moment of inertia of rotor
- $\omega$ = angular speed of rotor = $d\theta/dt$
- $L$ = constant associated with speed-related losses (damping, eddy current, friction)
- $T_{ext}$ = other external mechanical load
Motor Electrical Model

Electrical model:
\[ V_s = L_m \frac{di}{dt} + i.R_m + K_T.\omega \]

where:
- \( V_s \) = supply voltage
- \( L_m \) = motor coil inductance
- \( R_m \) = motor coil resistance
- \( \omega \) = rotor speed
- \( K_T.\omega \) = back-emf induced voltage
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
MC9S08SH4 Task and Resource Scheduling

- Input PWM
- Current sense
- Hall input

Flowchart:
- Duty cycle measurement
- Input speed setpoint measurement
- Motor Speed Control
- ADC
- Overcurrent detection
- LockedRotor detection
- Comparator
- Motor Speed Measurement
- Commutation detection
- Timer
- TPM
- Motor Drive Output
- Tach Output
Agenda

- Introduction to Brushless DC Motors
- Motor Electrical and Mechanical Model
- Motor Speed Control Hardware Design
- Motor Speed Control Software Tasks
- Motor Speed Control Challenges
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
H-Bridge (Driving Sequence)

Power supply

Motor

Hall sensor (digitized)

1 Electrical Cycle

Phase

(A) (B)

Hall sensor (digitized)

H1

H2

L1

L2

Commutation

(A) (B)
H-Bridge (Efficient Driving Sequence)

1 Electrical Cycle

Phase
Hall sensor (digitized)

H1
H2
L1
L2

(A)
(B)

Commutation
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
Reduce in-rush current

- In-rush current due to change in fan speed setpoint can cause sudden current surge
- External power supplies may trip due to current surge

Solutions:
- Setpoint – current RPM differential limit
- Setpoint ramping
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
Over-current limit

رصد: مطلوب للهواءات ذات التدفق العالي للحد من آثار الضرر على الأدوات الإلكترونية

رصد: الإرتفاع في التدفق الناتج عن القيود المادية على الحركة الأسطوانية

رصد: يمكن تنفيذه باستخدام مقاومة قياس التدفق المكونة وموصل التخمين السريع – متاح في MC9S08SH4
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
Speed Control Accuracy

- PID speed control steady state error
- Output speed measurement accuracy
- Input PWM duty cycle measurement accuracy
Speed Control Accuracy

- PID speed control steady state error
- Output speed measurement accuracy
- Input PWM duty cycle measurement accuracy
PI Closed-Loop Speed Control Model

\[ G(s) = K_p + \frac{K_i}{s} \]

Motor driving System \( H(s) \)

Input speed \( e \)

Output speed

PI Motor Speed Control
PI Closed-Loop Speed Control

- Velocity form of PI, also called “incremental” or “differential” PI is used to control motor speed.
- Derived by differentiation of the standard form.

- The velocity form of PI:
  \[ G'(s) = K_p \cdot s + K_i \]
PI Closed-Loop Speed Control Model (Velocity form)

\[ G'(s) = K_p s + K_i \]

Motor driving System \( H(s) \)

Input speed + \[ e \] \( G'(s) \) = \( K_p s + K_i \) \( u \) \( 1/s \) \( c \) \( H(s) \) Output speed

PI Motor Speed Control (Velocity form)
PI Speed Control Steady State Error

Steady state error occurs when non-zero $e$ results in zero $u$ output.

Caused by implementation of PI Control loop in integer arithmetic:

$Ki \times e < 1 = 0$

$\Rightarrow e < \frac{1}{Ki}$

Maximum steady state error = $\frac{1}{Ki}$
Speed Control Accuracy

- PID speed control steady state error
- Output speed measurement accuracy
- Input PWM duty cycle measurement accuracy
Output speed measurement accuracy

- Output speed measurement accuracy depends on MCU clock accuracy
- Crystal can be used to provide accurate clock – but this is expensive solution
- MC9S08SH4 can run on internal oscillator.
  - Factory-trimmed with typical accuracy of 1% over voltage and temperature range
Speed Control Accuracy

- PID speed control steady state error
- Output speed measurement accuracy
- Input PWM duty cycle measurement accuracy
Measuring Input PWM duty cycle

“Standard” Method of measuring PWM duty cycle

Step 1: Low pass filter to convert PWM to analog voltage where analog voltage level is proportional to the PWM duty cycle

Step 2: MCU measures analog voltage level using internal ADC

Step 3: MCU derives duty cycle by dividing measured ADC value by maximum ADC value.
Measuring Input PWM duty cycle

Problems with ADC method:

- Accuracy is affected by ground noise when motor is running – especially bad for high-current motor
- Low pass filter circuit subject to component value variation (e.g. 1% resistors) and temperature variations.
- ADC performance limitations (output resistance limits of low pass filter circuit, non-linearity errors, zero-scale errors, full-scale errors, quantization errors)

Conclusion:

- It is odd to measure a digital signal by conversion to analog signal to derive its digital value!
Measuring Input PWM duty cycle

Random Sampling Method of measuring PWM duty cycle

Step 1: Level translation circuit to convert PWM signal voltage level to MCU logic voltage level

Step 2: MCU samples PWM signal at its GPIO pin at random intervals to derive PWM duty cycle

Duty cycle = (Total number of ‘1’ samples) ÷ (Total number of samples)
Measuring Input PWM duty cycle

Mathematics behind random sampling:

- One reading of GPIO will yield either logic ‘1’ or ‘0’
- Therefore, one sample of PWM signal is a Bernoulli trial; where the probability of reading logic ‘1’ is:

\[ P(\text{reading a logic ‘1’}) = \text{duty cycle of PWM signal} \]

- For multiple independent samples, this becomes a binomial distribution
Measuring Input PWM duty cycle

According to Binomial Distribution:

- The proportion of samples with logic ‘1’ = \( np/n = p \)
- The standard deviation is \( (p(1-p)/n)^{\frac{1}{2}} \)
Measuring Input PWM duty cycle

When \( n \) is large, Central Limit Theorem states that the binomial distribution can be approximated by a Normal distribution.

Based on approximation to Normal distribution, the relationship between actual mean (\( p \)), observed mean (\( X \)), standard deviation (\( \sigma \)) and \( Z \), the standard normal distributed variable is:

\[
p = X \pm Z \sigma \approx X \pm Z \sqrt{(X(1-X)/n)}
\]

Hence, the probability that \( p \) is outside of above range is defined by the \( Z \), the standard normal distributed variable:

\[
1 - P((X - Z \sigma ) < p < (X + Z \sigma )) = 1 - P(-Z < z < Z)
\]
Error contributions between Input PWM and Output Speed (RPM):

- Input PWM duty cycle measurement error
- Output speed measurement error
  - E.g. MCU datasheet indicates MCU clock error less than 1% - this will result in up to 1% error in output speed measurement error
- PID Control Loop steady state error
  - E.g. if the Integral Gain is 0.0722, the maximum steady state error is \( \frac{1}{0.0722} = 13 \) (rounded down)
Speed Control Accuracy - Conclusion

Therefore, to achieve a certain output RPM error limit:

- Duty cycle measurement error + Output speed measurement error + PID steady state error $\leq$ Output RPM error limit
- Duty cycle measurement error $\leq$ Output RPM error limit - Output speed measurement error - PID steady state error
# Speed Control Accuracy - Conclusion

We can put all these into a spreadsheet:

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to obtain single duty-cycle measurement (s)</td>
<td>0.5</td>
<td>Implementation specific - dependent on MCU</td>
</tr>
<tr>
<td>Minimum operating speed (RPM)</td>
<td>900</td>
<td></td>
</tr>
<tr>
<td>Maximum operating (RPM)</td>
<td>4000</td>
<td>These parameters determine the fan operating curve (RPM vs. duty cycle)</td>
</tr>
<tr>
<td>Minimum duty cycle (%)</td>
<td>0%</td>
<td></td>
</tr>
<tr>
<td>Maximum duty cycle (%)</td>
<td>100%</td>
<td></td>
</tr>
<tr>
<td>PID Loop maximum steady state error (RPM)</td>
<td>17</td>
<td>This error is the reciprocal of the PID Integral Gain</td>
</tr>
<tr>
<td>Maximum MCU clock error (%)</td>
<td>1.00%</td>
<td>Obtained from MCU datasheet.</td>
</tr>
<tr>
<td>Maximum allowed RPM error (RPM)</td>
<td>50%</td>
<td>Specified by user/customer</td>
</tr>
<tr>
<td>Duty cycle (%)</td>
<td>50%</td>
<td>Operating Duty Cycle. Worst case error occurs at 50% duty cycle</td>
</tr>
<tr>
<td>Max Error @ specified duty cycle (%)</td>
<td>1.8000%</td>
<td>This must be less than &quot;Maximum allowed duty cycle error (%)&quot;; otherwise, background color will change to red</td>
</tr>
<tr>
<td>Number of random samples required to perform one duty cycle measurement</td>
<td>50000</td>
<td></td>
</tr>
</tbody>
</table>

| Computed Parameters | | |
|---------------------|------------------|
| Fan operating curve slope [A] | 3100 | Desired operating RPM = A*p + B |
| Fan operating curve offset [B] | 900 | |
| Maximum allowed error due to duty cycle error @ specified Duty Cycle (RPM) | 58.5 | |
| Maximum allowed duty cycle error (%) | 1.89% | |
| Standard normal variable limit | 8.049844719 | This limit determines the confidence interval that the actual duty cycle lies within the specified limits of the observed duty cycle. |
| Probability of one duty cycle measurement exceeding Max Error [Pe] | 8.8817842E-16 | "Failure" is the case when measured duty cycle deviates from actual duty cycle by more than Max Error |
| MTBF (hours) | 1.5637499E+11 | |
Motor Speed Control Challenges

- Phase commutation efficiency
- Reduce in-rush current
- Over-current limit
- Speed Control Accuracy
- Tuning motor speed curve
Tuning Motor Speed Curve

- Must tune quickly to meet customer changing requirements
- Complex motor speed curve requirements
- Avnet’s solution: Configuration by Excel spreadsheet!
  - Easy to use
  - Generates firmware automatically
## Tuning Motor Speed Curve

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data revision</td>
<td>0000</td>
<td>User should change this for each new set of parameters they use</td>
</tr>
<tr>
<td>Proportional gain</td>
<td>5000</td>
<td>Proportional gain of PI control. The actual value is a fraction of 2(^{16})</td>
</tr>
<tr>
<td>Integral gain</td>
<td>2500</td>
<td>Integral gain of PI control. The actual value is a fraction of 2(^{16})</td>
</tr>
<tr>
<td>Minimum input duty cycle (%)</td>
<td>95.00%</td>
<td>Minimum input duty cycle at which motor can start running</td>
</tr>
<tr>
<td>Maximum input duty cycle (%)</td>
<td>95.00%</td>
<td>Maximum input duty cycle at which motor runs as maximum speed</td>
</tr>
<tr>
<td>Minimum tach speed (RPM)</td>
<td>1400</td>
<td>Minimum speed at which motor should run</td>
</tr>
<tr>
<td>Maximum tach speed (RPM)</td>
<td>4600</td>
<td>Maximum speed at which motor should run</td>
</tr>
<tr>
<td>Lower hysteresis minimum input duty cycle (%)</td>
<td>11.00%</td>
<td>Lower hysteresis minimum limit at which motor will stop running</td>
</tr>
<tr>
<td>Upper hysteresis minimum input duty cycle (%)</td>
<td>15.00%</td>
<td>Upper hysteresis minimum limit at which motor will stop running</td>
</tr>
<tr>
<td>Lower hysteresis Full Speed input duty cycle (%)</td>
<td>97.00%</td>
<td>Lower hysteresis minimum limit at for full speed mode</td>
</tr>
<tr>
<td>Upper hysteresis Full Speed input duty cycle (%)</td>
<td>99.00%</td>
<td>Upper hysteresis minimum limit at for full speed mode</td>
</tr>
<tr>
<td>Lower hysteresis maximum input duty cycle (%)</td>
<td>93.00%</td>
<td>Lower hysteresis maximum limit at which motor will stop running</td>
</tr>
<tr>
<td>Upper hysteresis maximum input duty cycle (%)</td>
<td>96.00%</td>
<td>Upper hysteresis maximum limit at which motor will stop running</td>
</tr>
<tr>
<td>Locked motor restart time (seconds)</td>
<td>13</td>
<td>Delay in restarting motor after motor locked condition is detected</td>
</tr>
<tr>
<td>Maximum Full Speed limit (RPM)</td>
<td>6500</td>
<td>Maximum speed limit at full speed mode</td>
</tr>
</tbody>
</table>
### Tuning Motor Speed Curve

#### Excel Table

<table>
<thead>
<tr>
<th>Duty cycle (%)</th>
<th>Increasing duty cycle</th>
<th>Decreasing duty cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>23</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>25</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>26</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>28</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>29</td>
<td>29</td>
<td>29</td>
</tr>
<tr>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
</tbody>
</table>

#### Graph

- **Speed vs. Duty cycle**
- **Increasing cycle**
- **Decreasing cycle**