Driving a BLDC Motor using the HT45FM03B (Assembly Language)

D/N : AN0229E

Introduction

Brushless DC motors, in using electric commutation and PWM drive methods, greatly extend the life of electric motors, by overcoming problems derived from DC motor mechanical commutation. Applications in recent years have seen widespread use of such motors in areas such as home appliances, electric bikes, digital machine tools, robotics, model airplanes and so forth.

The HT45FM03B is a Holtek high performance RISC architecture MCU mainly applicable for use in brushless DC motor control applications. It contains an internal comparator, OPA, multi-channel AD converter, multi-channel external interrupt and dead time complementary PWM controller.

The following application uses the HT45FM03B as the cored device in a brushless DC motor control system, and describes the relevant functions of brushless motors such as the speed setting, commutating principles, PWM modulation, motor over-current protection and motor fast-braking methodologies.

Basic Features

The HT45FM03B is the core of the brushless DC motor control system and contains a host of fully integrated features such as:

- Operating voltage: $f_{SYS}=0.4\sim20\text{MHz}$ at $4.2V\sim5.5V$
- 26 bidirectional I/O lines
- External interrupt inputs shared with 4 I/O lines
- 8-bit programmable timer/event counter with overflow interrupt and 7-stage prescaler
- 16-bit programmable timer/event counter with overflow interrupt and 7-stage prescaler
- 4096x15 Flash Program Memory
- Flash Type MCU
- 192x8 Data Memory RAM
- Crystal, internal RC and external RC oscillators
- Fully integrated RC oscillator with three fixed frequencies: 12MHz, 16MHz and 20MHz
- Watchdog timer function
- PFD for audio frequency generation
Driving a BLDC Motor using the HT45FM03B (Assembly Language)

- HALT and wake-up functions to reduce power consumption
- Up to 0.2\(\mu\)s instruction cycle with 20MHz system frequency at \(V_{DD}=5\)V
- 8-level subroutine nesting
- 8-channel 12-bit resolution A/D converter
- 3 pairs of 10-bit PWM functions with complementary outputs shared with six I/O lines and with 3 PWM duty control registers.
- Bit manipulation instruction
- Table read instructions
- 63 powerful instructions
- All instructions in one or two machine cycles
- Low voltage reset function
- Low voltage protect function
- Operational Amplifier
- One comparator with interrupt function
- 28-pin SOP package (figure 1)

After explaining the basic features of the HT45FM03B, the following content will introduce the relevant components of the HT45FM03B for use in brushless DC motor control as well as the relevant control principles.

**BLDC Motor Introduction**

Brushless DC (BLDC) motors use semiconductor components to implement electrical commutation, which is a necessary aspect of DC motor driving. This replaces the usual mechanical commutator and electric brushes with semiconductor switching components. A brushless DC motor is composed of a permanent magnet rotor, multi-pole winding stator and position sensors etc. which together provide the motor with many advantages over conventional DC motors such as high reliability, no commutating sparking and low mechanical noise.
ADC Application in the BLDC Motor

Speed Setting

The speed setting value will transmit a voltage via an AD channel to the MCU. The MCU will use the converted value as base value for speed. As Figure 2 illustrates, VR is used to adjust the speed setting voltage value. R9 together with C15 form a low pass filter circuit.

![Figure 2](image)

A/D Converter Introduction

The related A/D converter registers in the HT45FM03B are ADRL, ADRH, ADCR and ACSR. The following content describes the relevant settings for these registers to implement an A/D conversion.

- **ADCR**
  - Select the required I/O pins for the A/D channel by setting the PCR0~PCR2 bits, which are bits 3~5 of the ADCR register.
  - Select a corresponding A/D pin to connect to the internal A/D converter by setting the ACS0~ACS2 bits, which are bits 0~2 in the ADCR register.
  - Set bit 7 (START bit) in the ADCR register from low to high and then low again (0 → 1 → 0), after which the AD conversion process will be initiated.
  - Monitor bit 6 (EOCB bit) of the ADCR register to check if it is “0” to determine if the A/D conversion process has completed.

- **ACSR**
  - The AD clock source is determined by the ADCS0~ADCS3 bits which are bits 0~4 in the ACSR register.

- **ADRH, ADRL**
  - This is where the AD conversion results are placed with the high 8-bits in the ADRH register and low 4-bits in the ADRL register.

The above is an introduction to the A/D converter in the HT45FM03B. For more details refer to datasheet and the A/D converter application notes on the Holtek website.
Hall Sensor Application in the BLDC Motor

Hall Sensor Detect Circuit in the HT45FM03B
As shown in figure 3, the three Hall Sensor lines are connected to INT0A~INT0C of external interrupt 0 in the HT45FM03B. Pins INT0A~INT0C are shared with PA4, PA5, and PA6. As long as a falling edge signal occurs from any INT0A, INT0B and INT0C line, an external interrupt 0 will be generated. Read the Hall Sensor status from the external interrupt 0 service program to implement mechanical commutation.

Presently each brushless DC motor is installed with three Hall Sensors. For the HT45FM03B, there are eight input status levels, which are 001, 010, 011, 100, 101, 110, 111 and 000, and which have a 120 degree rotational angle. Note that status levels 111 and 000 are not used. The function of Hall Sensor lies in executing not only the motor commutation but also for position and speed feedback. The advantage of using Hall Sensor feedback to measure speed is to reduce costs however may suffer in terms of positioning accuracy.

In figure 3, R1, R2 and R3 are pull-high resistors while R4, R5 and R6 are current limiting resistors. C7, C8 and C9 are bypass capacitors while D1, D2 and D3 are used as isolating diodes to prevent motor noise from coupling into the HT45FM03B.

Motor Commutation Principles
Brushless DC motor driving is based on using Hall Sensors to control six MOSFETs that commutate the motor and also drive the motor operation. Figure 4 illustrates the principles of using six MOSFETs to drive a motor.
For brushless DC motors, there are 60 and 120 degree phases while for each motor the corresponding status of the Hall Sensor and MOSFETs is as follows:

60 degree brushless motor commutation truth table:

<table>
<thead>
<tr>
<th>Hall sensor (SA : SB : SC)</th>
<th>MOSFET status</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Q1, Q4 on</td>
</tr>
<tr>
<td>001</td>
<td>Q5, Q4 on</td>
</tr>
<tr>
<td>011</td>
<td>Q5, Q2 on</td>
</tr>
<tr>
<td>111</td>
<td>Q3, Q2 on</td>
</tr>
<tr>
<td>110</td>
<td>Q3, Q6 on</td>
</tr>
<tr>
<td>100</td>
<td>Q1, Q6 on</td>
</tr>
</tbody>
</table>

120 degree brushless motor commutation truth table:

<table>
<thead>
<tr>
<th>Hall sensor (SA : SB : SC)</th>
<th>MOSFET status</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Q1, Q4 on</td>
</tr>
<tr>
<td>001</td>
<td>Q5, Q4 on</td>
</tr>
<tr>
<td>011</td>
<td>Q5, Q2 on</td>
</tr>
<tr>
<td>010</td>
<td>Q3, Q2 on</td>
</tr>
<tr>
<td>110</td>
<td>Q3, Q6 on</td>
</tr>
<tr>
<td>100</td>
<td>Q1, Q6 on</td>
</tr>
</tbody>
</table>

Quick Motor Brake Principle

The motor brake plays an important role in the positioning system. Many disadvantages exist with traditional mechanical brakes, such as increasing positioning error with the worn-down brakes. The E-ABS (the electromagnetic control brake) overcomes all these mechanical brake problems. The way of using the E-ABS electric brake is as follows:

- At least two phases may simultaneously open the three upper arms or lower arms through the HT45FM03B to make three motor phases a short circuit. Meanwhile the stator coil inside the motor will form a closed circuit. When the stator starts turning, the magnetic field will be cut and thus generate braking forces.

- The present brushless motor control method uses a three-phase six status design. The eight statuses (for 120 degree motors) of the three Hall Sensors include: 001, 010, 011, 100, 101, 110, 111 and 000, among which 000 and 111 are not used. The other six status conditions correspond to six positions of the motor. When the motor is driven using the Hall Sensor signal sequence of 001 → 010 → 011 → 100 → 101 → 110, the motor will rotate in a forward direction. When the motor is driven with an opposite Hall Sensor signal sequence, the motor will try to rotate in a backward direction and so generate a braking effect.
PWM Application in the BLDC Motor

HT45FM03B PWM Functional Description
The HT45FM03B provides three pairs of complementary PWM outputs shared with the PC0~PC5 pins. Each PWM is named PWM0H~PWM2H whose complementary ports are named PWM0L~PWM2L. The complementary PWM output will increase the operating efficiency of the motor.

A dead time function is included in the complementary PWM outputs in the HT45FM03B to avoid the possibility of virtual short circuits between high and low drives.

The HT45FM03B also provides an interrupt service routine entry for the PWM to generate an interrupt with the rising edge of every duty cycle.

The HT45FM03B provides four PWM operation modes, the 10-bit mode, (9+1)-bit mode, (8+2)-bit mode and (7+3)-bit mode, for different PWM outputs under different frequencies.

Functional Settings

- **PWM Clock Setting**
  The PWM clock is determined by bit 5 ~ bit 7 (PWMPS0~PWMPS2) in the PWMC1 register.
  The relation of the PWM clock $f_{PWM}$ and the system clock $f_{SYS}$ is shown in the table below:

<table>
<thead>
<tr>
<th>PWMPS2 : PWMPS1 : PWMPS0</th>
<th>PWM Clock</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>$f_{PWM} = f_{SYS}$</td>
</tr>
<tr>
<td>001</td>
<td>$f_{PWM} = f_{SYS}/2$</td>
</tr>
<tr>
<td>010</td>
<td>$f_{PWM} = f_{SYS}/3$</td>
</tr>
<tr>
<td>011</td>
<td>$f_{PWM} = f_{SYS}/4$</td>
</tr>
<tr>
<td>100</td>
<td>$f_{PWM} = f_{SYS}/5$</td>
</tr>
<tr>
<td>101</td>
<td>$f_{PWM} = f_{SYS}/6$</td>
</tr>
<tr>
<td>110</td>
<td>$f_{PWM} = f_{SYS}/7$</td>
</tr>
<tr>
<td>111</td>
<td>$f_{PWM} = f_{SYS}/8$</td>
</tr>
</tbody>
</table>

- **PWM Operating Mode Options**
  There are four PWM operating modes, 10-bit mode, (9+1)-bit mode, (8+2)-bit mode, (7+3)-bit mode, for different PWM outputs under different frequencies. The four modes can be chosen using configuration options.

- **Master Control for the PWM Outputs (PWMCTRL)**
  Bit 6 (PWMCTRL) in the PWMC0 register acts as the master control bit for the PWM outputs. When this bit is “1”, the PWM outputs will be active and when the bit is “0”, the PWM outputs will be placed into their inactive state.

- **PWM Output Selection Method (PWMCM)**
  Bit 1 (PWMCM) in the PWMC2 register selects which control method is used for the PWM outputs. When this bit is “1”, the PWM outputs are controlled by the PCPWMC register and when the bit is “0”, the PWM output will be controlled both by the PC6 and PC7 pins.

- **PWM Output Enable Control (PWMEN)**
  Bit 0 (PWMEN) in the PWMC0 register is the enable bit for the PWMxH outputs. When this bit is “1”, the PWMxH outputs will be enabled or be disabled when the bit is “0”.

- **PWM Complementary Output Enable Control (PWMCEN)**
  Bit 1 (PWMCEN) in the PWMC0 is the enable bit for the PWMxL outputs. When this bit is “1”, the PWMxL outputs will be enabled or disabled when the bit is “0”.


• **PWM Output Option Control Setting**

The PCPWMC is the PWM output option control register. The output control options are shown in figure 5:

![Figure 5 PCPWMC Register](image)

- b7 - b0
  - PWM output options
  - PC0: output mode
    - 0: I/O mode
  - PC1: output mode
    - 0: I/O mode
  - PC2: output mode
    - 0: I/O mode
  - PC3: output mode
    - 0: I/O mode
  - PC4: output mode
    - 0: I/O mode
  - PC5: output mode
    - 0: I/O mode
  - PWM buffer enable/disable
    - 0: PWM buffer enabled
    - 1: PWM waveform not updated until present PWM ends
  - PWM waveform updated immediately after present PWM subcycle

Not implemented, read as "0"

• **PWM Data Output Control Setting**

PCPWMD is the PWM data output control register, operating as shown in figure 6.

![Figure 6 PCPWMD Register](image)

- b7 - b0
  - PWM data output options
  - PWM0H on/off control
    - 0: inactive level
  - PWM0L on/off control
    - 0: inactive level
  - PWM1H on/off control
    - 0: inactive level
  - PWM1L on/off control
    - 0: inactive level
  - PWM2H on/off control
    - 0: inactive level
  - PWM2L on/off control
    - 0: inactive level

Not implemented, read as "0"
• PWM Stop Output Setting
The PWM stop output is controlled by Bits 2 ~ bit 4 (PWMSP0~PWMSP2) in the PWMC1 register. The control method is as follows:

<table>
<thead>
<tr>
<th>PWMSP2:PWMSP1:PWMSP0</th>
<th>PWM Stop Output Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>By software</td>
</tr>
<tr>
<td>001</td>
<td>By COUT / PA3 falling edge</td>
</tr>
<tr>
<td>010</td>
<td>By INT1 interrupt</td>
</tr>
<tr>
<td>011</td>
<td>By COUT / PA3 falling edge or INT1 interrupt</td>
</tr>
<tr>
<td>100</td>
<td>When a COUT / PA3 falling edge occurs, the PWMxH and PWMxL outputs are invalid. When a COUT / PA3 rising edge occurs, the PWMxH and PWMxL outputs are active.</td>
</tr>
<tr>
<td>101</td>
<td>When a COUT / PA3 falling edge occurs, the PWMxH output is invalid. When a COUT / PA3 rising edge occurs, the PWMxH output is active.</td>
</tr>
<tr>
<td>110</td>
<td>When a COUT / PA3 falling edge occurs, the PWMxH output is invalid.</td>
</tr>
<tr>
<td>111</td>
<td>When a COUT / PA3 falling edge occurs, the PWMxH and PWMxL outputs are invalid.</td>
</tr>
</tbody>
</table>

Dead Time Setting
• PWM Dead Time Output Enable Control - DTEN
Bit 2 (DTEN) in the PWMC0 register is the enable bit to determine whether the dead time function is to be added to the PWM complementary output. When this bit is “1”, the dead time function will be included in the PWM complementary output. When this bit is “0”, the dead time will be disabled.

• Dead Time Clock Option
The Dead Time Clock is determined using bits 6 and 7 (DTPS0, DTPS1) of the MISC register which selects a value for fD, as shown in the table below.

<table>
<thead>
<tr>
<th>DTPS1 : DTPS0</th>
<th>Dead Time Clock</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>fo = fSYS</td>
</tr>
<tr>
<td>01</td>
<td>fo = fSYS/2</td>
</tr>
<tr>
<td>10</td>
<td>fo = fSYS/4</td>
</tr>
<tr>
<td>11</td>
<td>fo = fSYS/8</td>
</tr>
</tbody>
</table>

• Dead Time Option
The Dead Time length option is selected using bits 3 ~ 5 (PWMDT0~PWMDT2) in the PWMC0 register. Details are shown in the following table:

<table>
<thead>
<tr>
<th>PWMDT2 : PWMDT1 : PWMDT0</th>
<th>Dead Time Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1/fo</td>
</tr>
<tr>
<td>001</td>
<td>2/fo</td>
</tr>
<tr>
<td>010</td>
<td>3/fo</td>
</tr>
<tr>
<td>011</td>
<td>4/fo</td>
</tr>
<tr>
<td>100</td>
<td>5/fo</td>
</tr>
<tr>
<td>101</td>
<td>6/fo</td>
</tr>
<tr>
<td>110</td>
<td>7/fo</td>
</tr>
<tr>
<td>111</td>
<td>8/fo</td>
</tr>
</tbody>
</table>
• Dead Time function for the complementary PWM outputs

The reason for including a dead time function in the PWM outputs in the HT45FM03B is to avoid the same set of high and low PWM drivers (PWMxH and PWMxL) from being activated simultaneously thus creating a virtual short circuit.

For example: the output waveform for PWMxH and PWMxL without a dead time is shown in figure 7.

![Figure 7](image1)

The problem with what is shown in figure 7 is that at position T1, the PWMxH bit becomes active and also the PWMxL bit also changes to inactive. At this time if the driver is quickly turned on but slowly turned off, the condition may arise where the PWMxH and PWMxL drivers are simultaneously active for a short time.

The output waveforms for PWMxH and PWMxL, including a dead time, is shown in figure 8.

![Figure 8](image2)

For the PWMxH and PWMxL outputs with dead time as shown in figure 8, the PWMxL bit changes to Inactive at T1 while the PWMxH changes to Active at T2. Now the condition where both a high and low driver are active at the same time can be avoided.

Some dead time options in the HT45FM03B should be noted:

- The Dead Time length should not be too short otherwise the both high and low may still be simultaneously active.
- The Dead Time length should not be too long to ensure good efficiency. A proper dead time length should be selected according to the speed of the selected driver.
PWMxH and PWMxL Output Logic Options

There are two selection bits in the PWM configuration options for the PWM output logic, the PWMLEV and PWMCLEV bits.

The PWMLEV bit is used to select the PWM0H~PWM2H bits that determine the logic connection of each PWM output modulation duty cycle.
If the PWMLEV bit is selected as Active High, then the value in PWMnH/PWMnL will determine the high level output width of the PWM modulation duty cycle.
If the PWMLEV bit is set as Active Low, then value in PWMnH/PWMnL will determine the low level output length of the PWM output modulation duty cycle.

For example: When the PWM operates in the (9+1) bits mode with fPWM = 8MHz, if the PWMLEV bit is selected as Active High and PWMH/PWML=0014H, the PWM waveform will be:

![Figure 9](image9.png)

If the PWMLEV bit is selected to be Active Low and PWMH/PWML=0014H, the PWM output waveform will be:

![Figure 10](image10.png)
PWM Data Output

The data bit assignment is in the PWM modulation data register, PWMnH and PWMnL, shown in Table 11:

<table>
<thead>
<tr>
<th>Register</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWMnH</td>
<td>D9</td>
<td>D8</td>
<td>D7</td>
<td>D6</td>
<td>D5</td>
<td>D4</td>
<td>D3</td>
<td>D2</td>
</tr>
<tr>
<td>PWMnL</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>D1</td>
<td>D0</td>
</tr>
</tbody>
</table>

Figure 11

The value in the PWM data register determines the duty ratio of the high/low level in one PWM modulation duty cycle.

PWM Control

In the PWM configuration options, the PWM on/off control can be setup by the PC0~PC5 registers. When PC0~PC5 is “1”, the PWM outputs will be enabled, or else disabled when the values are “0”. For example:

When PC0 = 1, the PWM0H output will be on; when PC0 = 0, the PWM0H output will be off.

PWM Interrupt Control

The PWM will generate an interrupt during every modulation duty cycle. The four PWM operating modes in the HT45FM03B are 10-bit mode, (9+1) bit mode, (8+2) bit mode and (7+3) bit mode. No matter what mode the PWM output is in, the PWM interrupt interval is fixed with the time of the PWM modulation duty cycle. Understanding the length of an PWM interrupt interval is very important when writing the PWM interrupt service program. The following example describes the way to calculate a PWM interrupt interval.

Example: When the HT45FM03B PWM is in the (9+1) bit mode, the calculation method of the PWM interval is:

\[ T = \frac{512}{f_{PWM}} \]

\( T \) means the length of an interrupt interval. \( f_{PWM} \) is the PWM clock frequency from the system frequency \( f_{SYS} \) divided by bit 5 ~ bit 7 (PWMPS0~PWMPS2) in the PWMC1 register.

If \( f_{SYS} = 8\text{MHz} \) and Bit 5 ~ bit 7 in the PWMC1 register is 001, \( f_{PWM} = \frac{f_{SYS}}{2} = 4\text{MHz} \) and \( T = \frac{512}{f_{PWM}} = \frac{512}{4\text{MHz}} = 128\mu\text{s} \)

In this condition, the PWM will generate an interrupt with a 128\( \mu \)s time interval.
MOSFET PWM Driving Principles

There are many methods to control the speed of a motor using MOSFETs, for example, using the IR2103 or the HT45B0C driver module. For a better understanding about driving principles, a drive circuit composed of transistors is shown in figure 12.

![Diagram of MOSFET PWM driving circuits](image)

**Figure 12**

In figure 12, the COIL is a motor winding and the MOSFET driver is divided into an upper arm drive circuit and a lower arm drive circuit. The reason why the upper arm drive circuit is more complicated than the lower arm circuit is because when both the upper and lower arms are conducting, the DC36V voltage will be switched down to the coil. Therefore D4 and D5 need to be added to the upper arm to isolate the diode, and C19 to maintain the capacitance.

Two functions are implemented in the driver circuit:

- Complete level transition with all switching circuits on the upper and lower arms. On the upper arm, the switching circuits are composed of R31, Q7, Q8 and R32 while the lower arm is composed of R35, Q10, Q11 and R37.

- The driver circuit should include a discharge circuit. This is needed for the case where the PWM quickly turns on and off as without this discharge circuit, the MOSFETs may not turn off completely due to the stray capacitance that exist on the MOSFET gates. The discharge circuit in the upper arm is composed of Q9 and R34 while the lower arm is composed of Q12 and R38.
Successive Approximation PWM Modulation Algorithm

Many methods can be used to control a motor using PWM, such as PI modulation and so on. Here a successive approximation method is shown as follows.

- Step 1: Read the speed data and set the AD value from which the corresponding PWM output value can be checked using a table read function. The PWM output value corresponds successively with speed.

- Step 2: In the PWM interrupt service routine, compare the current PWM output value with the selected PWM value at every interval. If the current PWM output value is smaller than the selected PWM value, increase the PWM output value, or reduce it if the output value is larger than the selected value until the two values are equal to each other. In this say, the speed will smoothly increase or decrease when executing speed control.

In short, the AD value determines the required speed value. To achieve the required speed, it is not just a case of transmitting the corresponding PWM value immediately but rather to successively increase or decrease the PWM output value in the PWM interrupt service routine to gradually reach the required speed.

BLDC Motor Operational Amplifier and Comparator Application

HT45FM03B CP Features

The HT45FM03B provides a comparator whose I/O pins are pin-shared respectively with PA1, PA2, and PA3 as shown in figure 13:

![Figure 13](image_url)

Comparator Register Setting

- CP Enable Control
  Bit 3 (CMPEN) in the MISC register is the comparator enable bit. When this bit is “0”, the comparator is disabled. When this bit is “1”, the comparator is enabled.

- CP Output Selection Control
  Bit 5 (COUTEN) in the MISC register is the comparator output selection bit. When this bit is “0”, PA3/COUT is used as a normal I/O. When this bit is “1”, the PA3/COUT is used as a comparator output.

- CP Delay Enable
  Bit 7 (CHYSON) in the CMPC register is the comparator delay enable bit. When this bit is “0”, the delay function will be disabled or else enabled when this bit is “1”.

- Comparator Offset Voltage Calibration Mode Selection
  Bit 6 (COFM) in the CMPC register is the comparator offset voltage calibration mode selection control bit. When this bit is “0”, it is used as a normal comparator mode; when this bit is “1”, it is used as the offset voltage calibration Mode.
Driving a BLDC Motor using the HT45FM03B (Assembly Language)

- Comparator Offset Voltage Calibration Input Selection
  Bit 5 (CRS) in the CPMC register is the comparator offset voltage calibration input selection control bit. When this bit is “0”, choose CVINN as the reference voltage input or choose CVINP as the reference voltage input when this bit is “1”.

- Comparator Offset Voltage Adjustment Control
  Bit 0~bit 4 (COF0~COF4) in the CMPC register is the comparator offset voltage adjust control bit.

- Comparator Interrupt Delay Time Setting
  The comparator interrupt delay time is decided by Bit 0~bit 3 (CMPDB0~CMPDB3) in the DBTC register with the corresponding reference delay time shown in the following table.

<table>
<thead>
<tr>
<th>CMPDB3 : CMPDB2 : CMPDB1 : CMPDB0</th>
<th>Delay Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0</td>
</tr>
<tr>
<td>0001</td>
<td>4/fSYS</td>
</tr>
<tr>
<td>0010</td>
<td>8/fSYS</td>
</tr>
<tr>
<td>0011</td>
<td>16/fSYS</td>
</tr>
<tr>
<td>0100</td>
<td>32/fSYS</td>
</tr>
<tr>
<td>0101</td>
<td>64/fSYS</td>
</tr>
<tr>
<td>0110</td>
<td>128/fSYS</td>
</tr>
<tr>
<td>0111</td>
<td>256/fSYS</td>
</tr>
<tr>
<td>1000</td>
<td>512/fSYS</td>
</tr>
<tr>
<td>1001~1111</td>
<td>1024/fSYS</td>
</tr>
</tbody>
</table>

Comparator Offset Voltage Calibration Steps
- Step 1: Set COFM to “1” to select the offset voltage calibration mode – here S3 is closed.
- Step 2: Set CRS to “0” or “1” to select whether CVINN or CVINP is the reference voltage.
- Step 3: Adjust COF0~COF4 until the COUT output voltage status changes and record the values of COF0~COF4.
- Step 4: Set COFM to “0” to select the normal comparator mode.

The above content is a brief introduction to the HT45FM03B comparator. For detailed information, refer to the datasheet and other comparator related application notes in Holtek’s official website.

HT45FM03B Operational Amplifier Features

The HT45FM03B provides an operational amplifier in which the OPVINP, OPVINN, and OPOUT are pin-shared with PA0, PB3 and PB2. The related settings are as follows.

- OPA Enable Control
  Bit 7 (OPAEN) in the OPAC register is the operational amplifier enable control bit. When this bit is “0”, the OPA is disabled. When this bit is “1”, the OPA is enabled.

- OPA Offset Voltage Calibration Mode Selection
  Bit 5 (AOFM) in the OPAC register is the OPA offset voltage calibration mode select bit. When this bit is “0”, it is used in the normal OPA mode. When this bit is “1”, it is used in the offset voltage calibration mode.

- OPA Offset Voltage Calibration Input Selection
  Bit 4 (ARS) in the OPAC register is the offset voltage calibration input select bit. When this bit is “0”, OPVINN is chosen as the reference voltage input. When this bit is “1”, OPVINP is chosen as the voltage input.
- OPA Offset Voltage Calibration Control
  Bit 0~bit 3 (AOF0~AOF4) in the OPAC register is the OPA offset voltage calibration control bit.

The above content is a brief introduction to the HT45FM03B operational amplifier. For detailed information, refer to the datasheet and other OPA related application notes on Holtek's official website.

**BLDC Motor OPA and Comparator Application**

The HT45FM03B includes an internal comparator and an operational amplifier. These are used to implement overcurrent protection for the motor. The comparator can offer a protection mechanism for momentary overcurrent and the operational amplifier can be used to monitor the operation current of the motor. The sampling current value can be acquired from a Constantan wire sensor.

![Diagram](image)

R23 in figure 14 is a constantan resistor of which the value corresponds to the motor power consumption. Using the internal OPA in the HT45FM03B to amplify the current, R24 and C16 will compose a current filter circuit that connects to the same pin of the OPA and uses R25 as the feedback resistor. In the HT45FM03B, the OPA output, being connected to the AD converter, allows the motor operation condition to be monitored through monitoring the current.

The other current filter circuit is composed using R29 and C17 and is connected to the negative terminal of the internal comparator in the HT45FM03B. R27 and R28, are connected to the positive terminal of the comparator, and are used to setup a standard value for the current. In the HT45FM03B, the comparator output is connected to the interrupt system.
When the current value is larger than the standard value, the comparator will switch quickly and generate an interrupt that can be used for protection handling.

Circuit Principles
Software Flowchart

Program Description

The main program starts by initializing the IO, RAM, SFR and the 4ms timing setting, and then follows with successive subroutine calls for button detection, speed modulator voltage detection, current detection, and speed control.

The current detect subroutine uses the A/D converter, which measures the current. This is obtained by first changing the measurement to a voltage and then amplifying. Save the detected voltage value in current_value and compare with the standard value in current_ref. If the current_value is larger than current_ref, set op_flag to "1" or else set to "0". After a delay of 30s, if current_value is still larger than current_ref, namely op_flag is "1", set over_flag to "1" otherwise set to "0".

The speed control subroutine is used to control the motor operation mainly by judging if the three flags, system_flag, cp_flag and over_flag, satisfy the motor operating conditions.
Sample Program

configuration option:

- OSC: external XTAL
- PWM mode: (9+1) bits mode
- Comparator interrupt source: Comparator output falling edge
- PWM duty mode: 1 PWM duty mode
- INT0A pin-shared option: Pin share with PA4
- INT0B pin-shared option: Pin share with PA5
- INT0C pin-shared option: Pin share with PA6
- PD0/PFD: PD0
- PDI/RESB: RESB
- WDT: Enable

; other options selected by user.

For program and description, see the attachment.

Conclusion

The content above has introduced the functional components of the HT45FM03B for application in brushless DC motors with reference circuits and software programs. The HT45FM03B has been widely applied for use in electric bikes and in the remote control modeling application area.