Application note FMod-I2CDCMOT 48/1.5 Microcontroller interface

The FMod-I2CDCMOT 48/1.5 board can be controled directly from a standard microcontroler equiped with an I2C serial interface. FiveCo recommands the use of an FMod-TCP DB or FMod-TCP BOX board to configure and test the FMod-I2CDCMOT 48/1.5, but it is possible to skip this step and to configure directly the board from the microcontroller. This document describes this.

Motor example

Let's use a 24V motor as example, with a maximum continuous current of 1 A, and a quadrature-coder of 500 CPRs.

500 CPRs (cycle per revolution) allows us to know 2000 (500 x4) different states for a single motor rotation = 2000 PPRs (pulses per revolution).

When this motor runs at 3000 rpm = 50 rps, our module counts 50 rps x 2000 (pulses/r) = 10000 (pulses/sec).

Step 1 – Setting PID and movement parameters

1. **Disconnect the motor from its mechanical components**.
2. Write register CURRENTMAX (0x2A), with the value corresponding to the current which will be needed for the application. In our example, the value to be written will be 0x00010000 (1 A).
3. Write 0x01 into the AUTOTUNING register (0x39) in order to start the automated parameters query.
   - The motor will run for about 2 seconds.
   - Read this register many times and wait until it either comes back to 0 (OK) or 4 (error).

   One can also wait 4 seconds to make sure the autotuning is completed.
4. Save all the parameters that have just been modified by using the SAVEUSERPARAMETERS function (0x03).

Step 2 – Position regulation test

This should only be done after the configuration of the PID and movement parameters have been completed.

1. (Optional) Write 0 into the POSITION register (0x26). Thus the motor's current position (while stopped) will be set to 0 (pulses).
2. Write 2000 (0x000007D0) into the INPUT register (0x21). This will be the new goal to be reached (in pulses).
3. Write 0x05 into the REGULATIONMODE register (0x20), in order to set the module to the "position with trajectory tracking" mode.

   The motor starts running and stops at 2000. Based on our example, 2000 is the equivalent of one motor rotation.
4. (Optional) to set a new goal (pulses), write the new target value into the INPUT register (0x21).

   It is not necessary to go through the configuration of the REGULATIONMODE register, which is already set to the "position with trajectory tracking" mode.

Step 3 – Speed regulation test

This should only be done after the configuration of the PID and movement parameters have been completed.

1. Write 10000 (0x00002710) into the INPUT register (0x21). This will be the target speed (pulses/sec).
2. Write 0x04 to the REGULATIONMODE register (0x20) in order to set the module to speed mode.

   The motor will run at a speed of 10000 pulses/sec (=3000 rpm based on our example).
3. (Optional) to set a new speed target, write the new value into the INPUT register (0x21). It is not necessary to go through the configuration of the REGULATIONMODE register, which is already set to the speed mode.

Step 4 – Setting of homing position (Homing)

There are many possible ways to set the homing position, but we use the most widespread one.

1. To one end of the mechanical movement, we attach a limit sensor (5V compatible): either a switch, an optical barrier or an inductive sensor.
2. If the sensor needs to be powered, it must be directly plugged on the FMod-I2CDCMOT 48/1.5; the 0v and 5v must be connected to the module. If the sensor doesn't need to be powered (switch), connect one of its terminals to the 5V of the FMod-I2CDCMOT 48/1.5.
3. The sensor's return line must be linked to the Limit I (LI-Home) of the FMod-I2CDCMOT 48/1.5

Ch. de la Rueyre 116 / 1020 Renens / Switzerland
Tel +41 21 6326010 / Fax +41 21 6326011

fiveco innovative engineering

www.fiveco.ch
info@fiveco.ch

4. Look at the sensor's documentation, and make sure which output signal type is used. There is quite often only one transistor (bipolar or MOS).
5. If the output transistor is a PNP-bipolar or a MOS-P, one must use a resistor defining the 0V state = pull-down resistor. If the output transistor is a NPN-bipolar or a MOS-N, one must use a resistor defining the 5V state = pull-up resistor.
6. Pull-ups and pull-downs already exist on the FMod-I2CDCMOT 48/1.5, but you will need to decide which one will be activated. Read the OPTIONS register (0x2C) and change bit 5 to value 1 in order to activate the pull-up. Set to value 0 to activate the pull-down.
7. Now, you must figure whether your sensor is active when the output is set to 0v (0) or 5v (1)?

   If by reading the documentation that came with your sensor you are still unsure, use a voltmeter and measure the voltage when your device is at the sensor position. If you measure 5v, your sensor will be active when reading 1. If you measure 0v, your sensor will be active when reading 0.

   In order to make sure that your system is correctly wired, take your device away from the sensor. The signal MUST be the opposite of what it was!
8. Write the LIMITSETUP register (0x50) with bit 0 equal to value 1, and bit 1 set to the value defined under last point.
9. When your device is away from the LI sensor, read the POSITION register. Read it again when the device is moved to the sensor's position: is it higher or lower than the first reading?

   This will define the Homing mode, with limit I at either LI + or LI -.
10. Write the HOMINGOPTIONS register (0x48) with the value 0x00BB4B0y, y should be set as y=8 for LI-, and y=9 for LI+
11. Decide which value will be assigned to the position once LI has been turned on for homing, and write this data to the HOMINGPOSITION register (0x4B).
12. As soon as the system has been calibrated, where should it go?

    Decide which position you would like it to be at and write it to HOMINGINPUT. If you have no specific idea, simply write the same value as the one indicated under last point).
13. Save all the previously modified parameters through the SAVEUSERPARAMETERS function (0x03) and wait ~0.5 seconds for the settings to be saved in EEPROM.
14. Everything is now ready for the homing through the HOMING function (0x49).

    The motor will run until the limit LI-Home is activated, reference is then recorded, and the system goes back to its stand-by position.

Author: Xavier Greppin 11.2006

Revision 1.0

Ch. de la Rueyre 116 / 1020 Renens / Switzerland
Tel +41 21 6326010 / Fax +41 21 6326011

www.fiveco.ch
info@fiveco.ch