MERITEK
Automation reliable & smart

MBs-CBCANH

Operation Manual

Frequency Converter Control – ABB example

Index

Figure index

Table index

Version	Date	Author	Description
V1.0	2017/05/24	Curtis Li	Draft
V1.1	2017/07/18	Curtis Li	Revised function block
V1.2	2017/07/19	Edison Lin	English version
V1.3	2017/08/04	Edison Lin	Modified the block ladders and example
V1.4	2017/11/01	Curtis Li	Add SDO task and NMT Task

1. Document purpose

This manual aims to provide a quick follow-through guide for using CBCANH to control a frequency converter. The well-known ABB is used as the reference example.

2. System Configuration

2.1 EDS

Use Load EDS button to import the electronic data sheet (EDS) for a specific slave device. Multiple imports are supported.

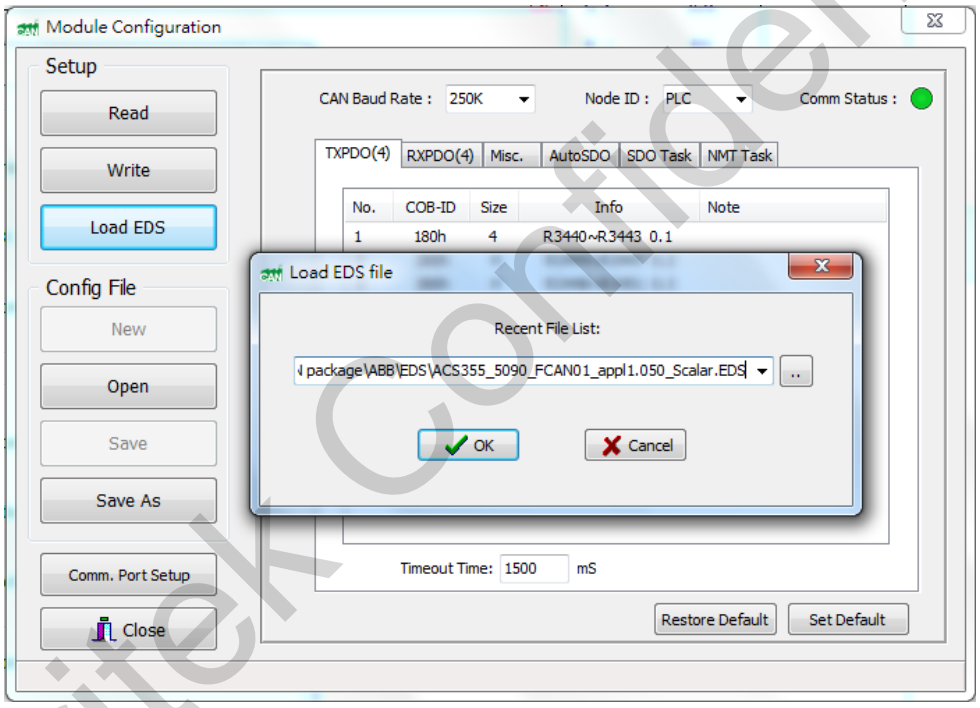


Figure 1 Load EDS

2.2 File operations

Use Open, Save, and Save As button to work on the CBCANH configuration file for the convenience of duplication and maintenance.

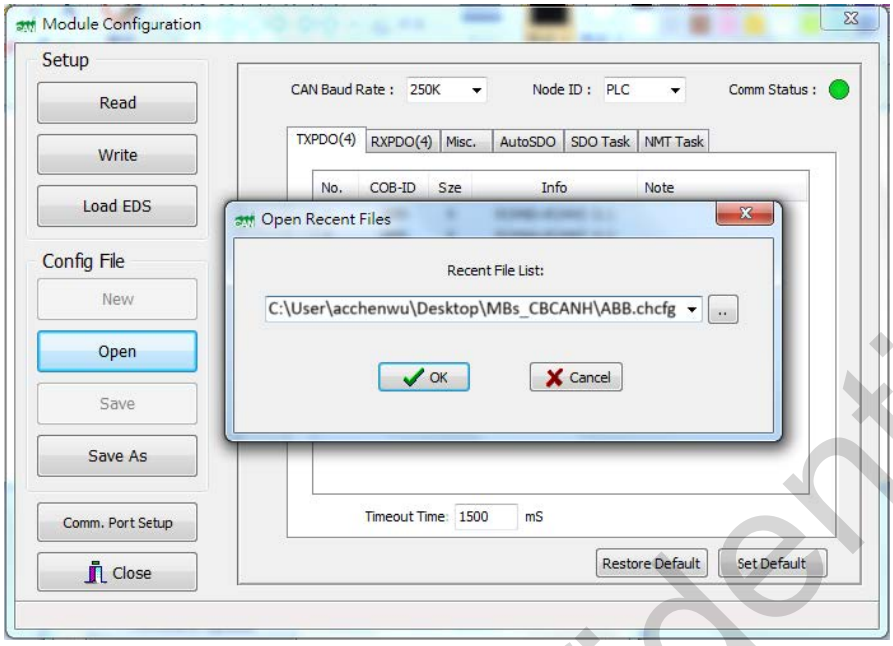


Figure 2 Import from a chcfg configuration file

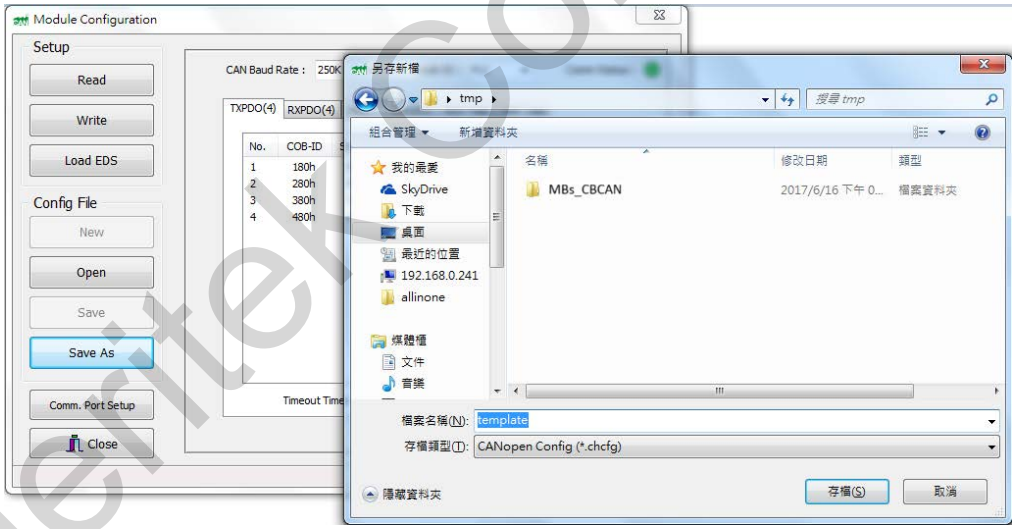


Figure 3 Export to a chcfg configuration file

2.3 Current configuration operations

Use Read button to read the current configuration from the CBCANH for further reviewing or editing; Use Write button to save the configuration result to the CBCANH.

2.4 AutoSDO

Maximum 30 groups are supported, each of which can be configured up to 12 SDO operations. Each group can have a unique node ID or share the same node ID if more operations are needed. AutoSDO will be executed sequentially during power on or through the block ladder AUTOSDO_CTRL provided by Meritek.

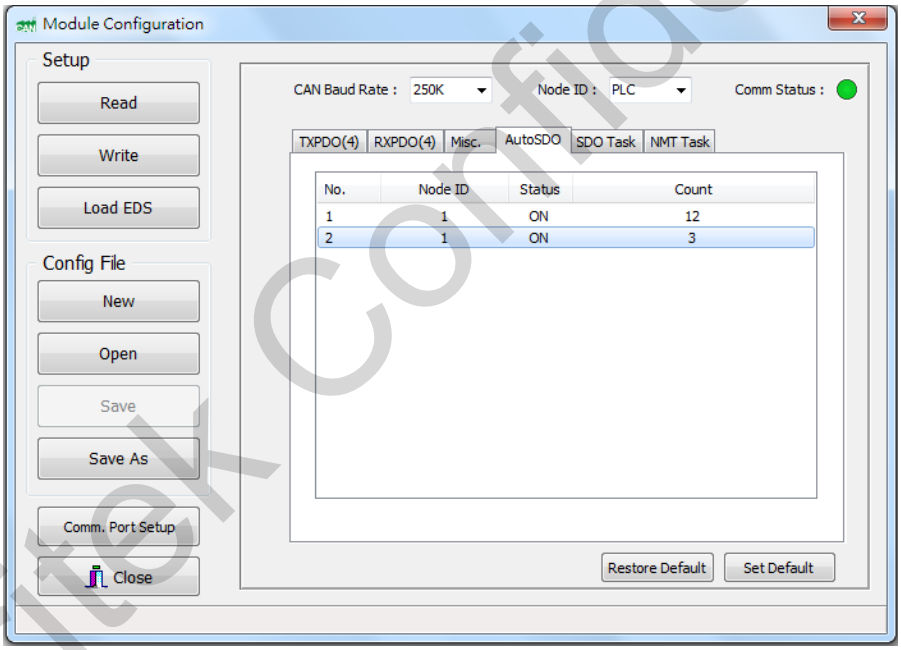


Figure 4 Different groups with the same node ID is possible

AutoSDO supports two modes, including WR(write) and MR(monitor). WR is a SDO operation which writes a given length of data (8/16/32 bits) into a slave device, e.g. PDO mapping or any preset value. MR is a SDO operation which reads data from a slave device and compares it with the expected data. A

mask will be used to do a logical AND with the data read, which makes the bit comparison possible.

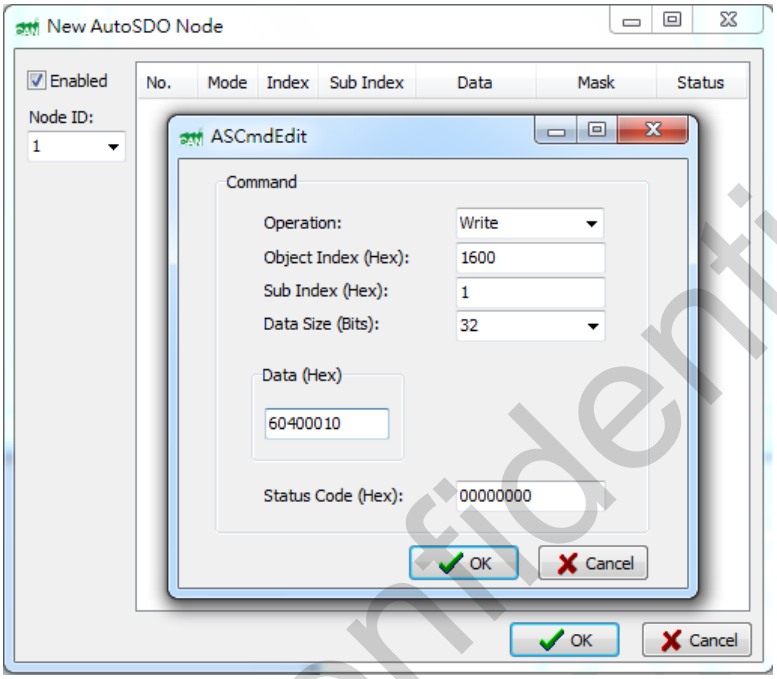


Figure 5 AutoSDO WR setup

MR is a SDO operation which monitors the value of a specific object dictionary index. As shown in the Figure 6, the following equation must satisfy for a monitoring operation to succeed.

LOGICAL_AND(UPLOADED DATA, 000FH) == 0007H

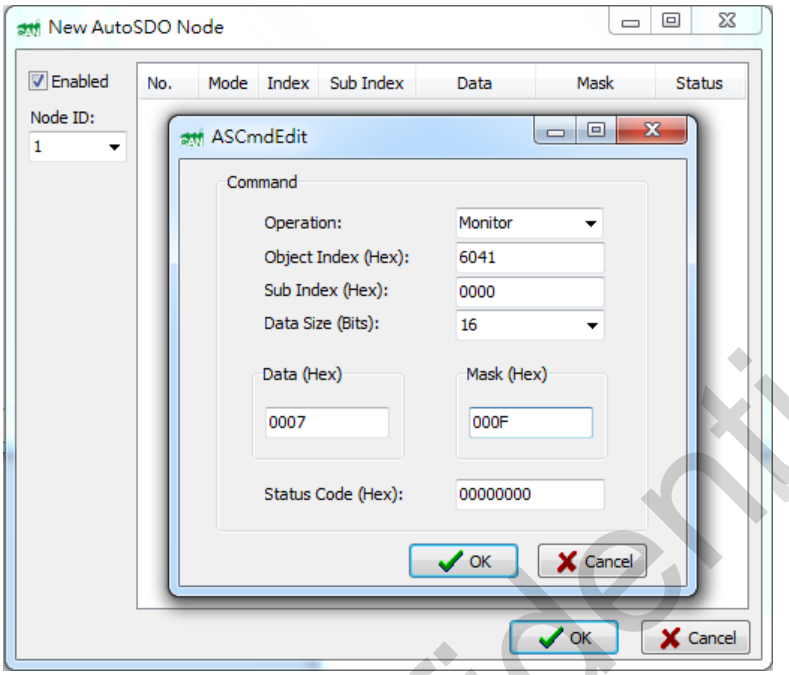


Figure 6 AutoSDO MR setup

2.5 SDO Task

SDO task makes it possible that SDO operations can be done by accessing registers of PLC. Adding a new SDO task is completed, by configuring it with index and sub-index of a specific node, operation mode, type and start address of corresponding PLC registers, in SDO task page. After the setup is finished, accessing the corresponding PLC registers is the same as accessing SDO data. Maximum 32 operations are supported.

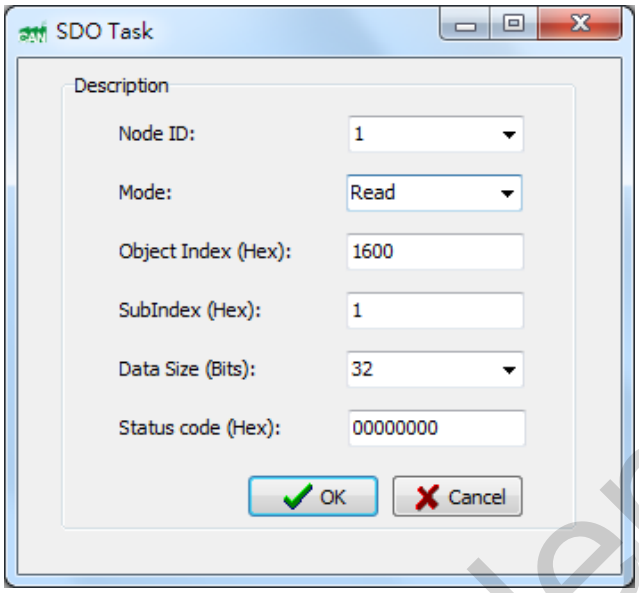


Figure 7 SDO task setup

As shown in Figure 7, SDO task supports two modes, including read and write. Both of them support data access in three variant data length (8/16/32 bits). Status code shows the result of execution, either success or error code is returned.

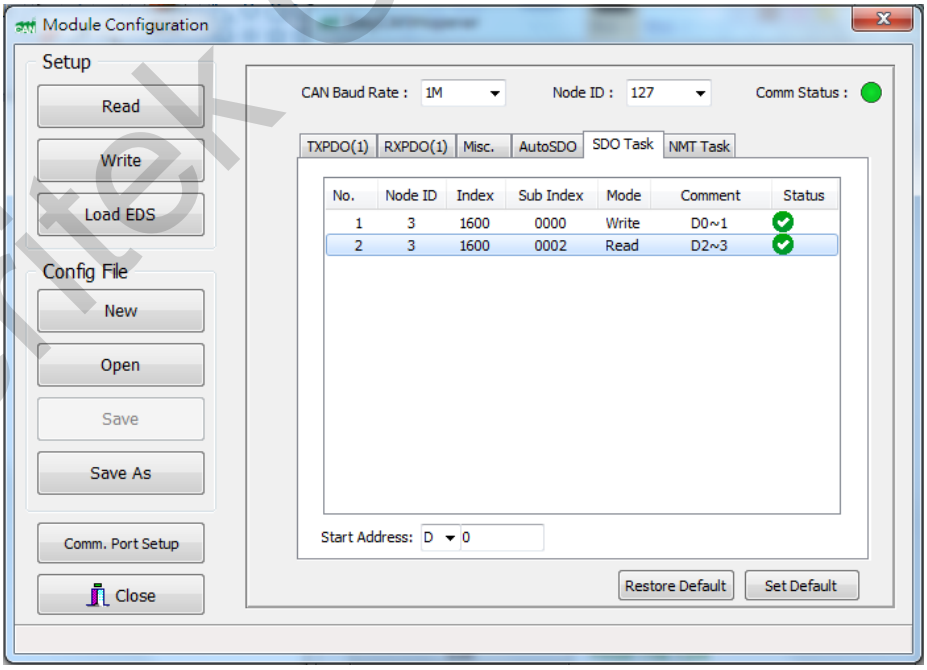


Figure 8 SDO task page

If the operation mode is read, the data accessed from slaves is put into the corresponding PLC registers. If the operation mode is write, the data which is about to be transmitted is put into the corresponding PLC registers. As shown in Figure 8, each SDO task occupies two PLC registers regardless of the data size.

2.6 NMT Task

NMT task makes it possible that NMT commands can be done by accessing registers of PLC. Adding a new NMT task is completed, by configuring it with target node, NMT command, type and start address of corresponding PLC registers, in NMT task page. After the setup is finished, accessing the corresponding PLC registers is the same as executing NMT command. Maximum 32 operations are supported

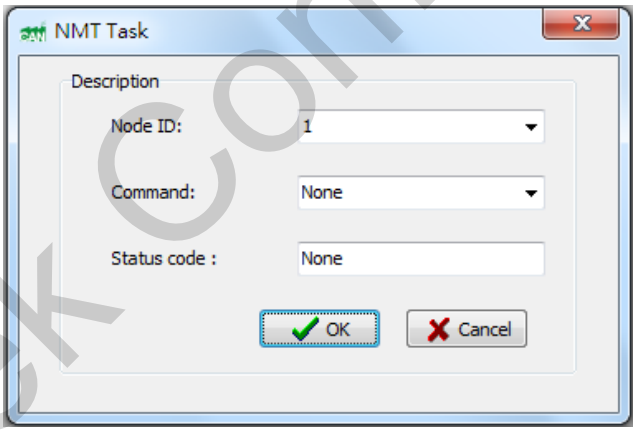


Figure 9 NMT task setup

The NMT task setup page is shown in Figure 9. Besides the same NMT commands that NMT Services has, NMT task has one additional command named “none”. NMT command “none” has no default command and used in the situation when the operation is decided sometime later. Status code shows the result of execution, either success or error code is returned.

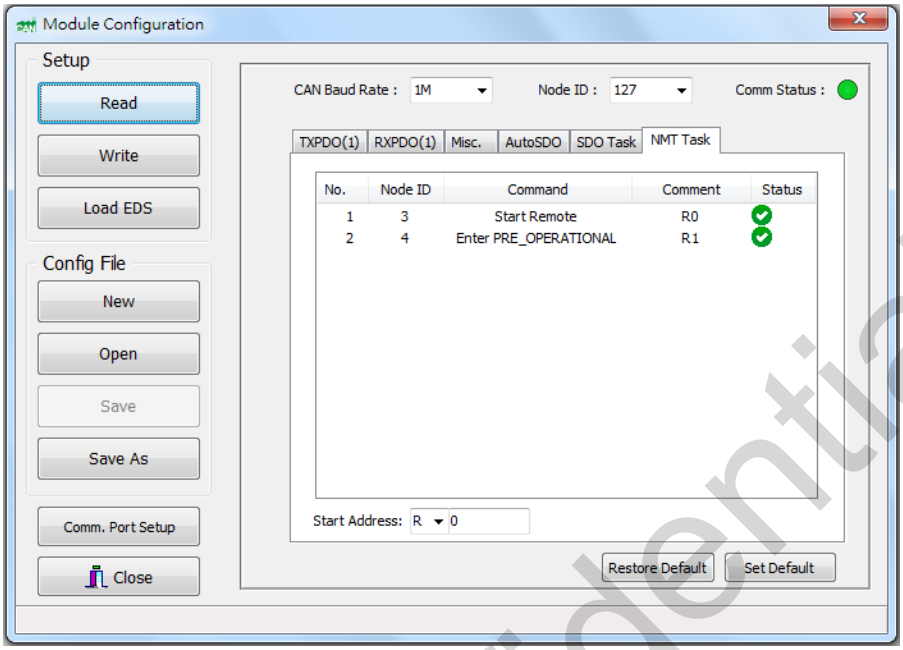


Figure 10 NMT task page

As shown in Figure 10, each NMT task occupies one PLC register. The Most Significant Byte(MSB) of corresponding PLC register stores NMT command, while the Least Significant Byte(LSB) of corresponding PLC register is used to trigger command and store the result of execution. The corresponding value of NMT command, as shown in Table 1, is filled into MSB. The value filled into LSB to trigger command is shown in Table 2.

NMT Command	Value
None	0
Start Remote	1
Enter Pre-Operational	2
Reset Node	3
Reset Communication	4
Stop	5

Table 1 Corresponding values of NMT commands

Status Code	Description
0001h	Successful
0002h	Failed
Other Values	Trigger Command

Table 2 Corresponding values of status codes

For example, to execute NMT command “Start Remote” through NMT task, one should set the content of corresponding PLC register as value shown below:

0103H

After execution, status code returns the result. If it works successfully, the content of corresponding PLC register should be the same as value shown below:

0101H

2.7 PDO setup

Refer to section 6.2.1.2 and 6.2.1.3 in the CBCANH user manual.

2.8 SYNC time

Refer to section 6.2.1.4 in the CBCANH user manual.

2.9 Auto. Start Remote

Refer to section 6.2.1.4 in the CBCANH user manual

3. Ladder program design notes

1) Use provided block ladders for CBCANH control
   - AUTOSDO_CTRL

As shown in Figure 11, it is a calling block which provides an alternative way to execute AutoSDO groups in the ladder program. As shown in Figure 12, it could be wrapped in a sub-function in order to be triggered in the program.

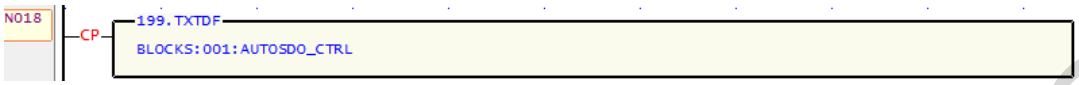


Figure 11 CBCANH-specific block ladder - AUTOSDO_CTRL

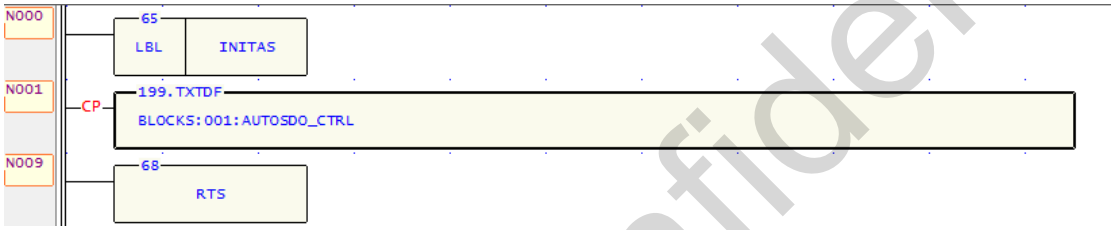


Figure 12 Sub-function including AUTOSDO_CTRL block ladder

- CMR

As shown in Figure 13, it is a block ladder used to update a certain set of PLC registers which maps to configured RPDOs to the CBCANH. Place only one of it in the bottom of the main ladder program and make sure the corresponding reserved registers have been set as intended. The number of registers to be transmitted is filled into R3116, and the start address is filled into R3106.

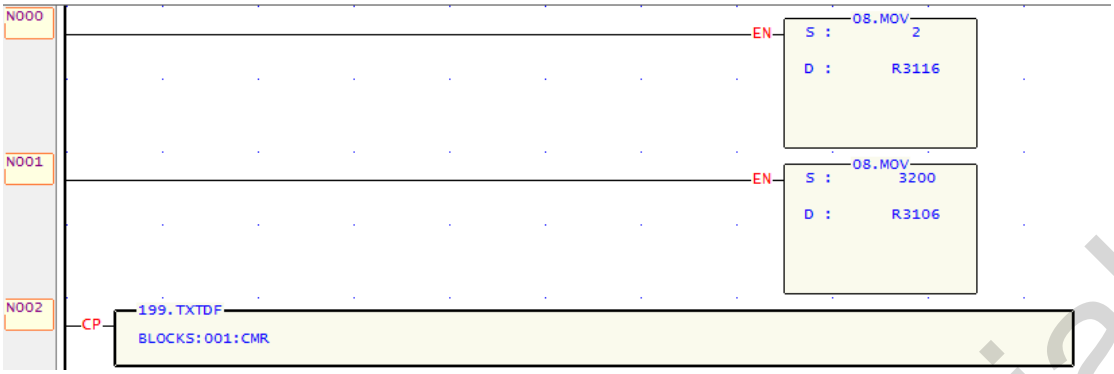


Figure 13 CBCANH-specific block ladder – CMR

2) A drive can be put into the Operation enabled state using AutoSDO, but be careful not to let a RPDO which maps to the control word(0x6040) affects the state and lead to an undesired initiation result. Use of the reserved PLC register M1924 is recommended for initialization when boot up.

4. Velocity control of a ABB frequency converter

4.1 Overview

This example has two parts. First part demonstrates how to turn on a motor by loading EDS to get pre-defined PDO configuration. Second part demonstrates how to control the velocity of motor by changing pre-defined PDO configuration by accessing the object dictionary. The setup overview is shown in Table 3.

Configuration item	Status
AutoSDO group	1
AutoSDO operations	3
EDS	Yes
TPDO	1 for status word
RPDO	1 for velocity and control word
CBCANH node ID	127

ABB node ID	3
Communication baud	1Mbps

Table 3 Setup overview in example

4.2 Turn on a motor

4.2.1 ABB frequency converter configuration

Refer to the velocity control setup in the ABB frequency converter user manual in p.61. To demonstrate the function of SDO task and NMT task, the pre-defined PDO configuration in the object dictionary is used in this example. The drive parameter 5104 is set to 0. Refer to the user manual for more details. The PDO mapping should be the same as shown in Figure 14.

PDO	Word 1
Rx PDO1	6040h Control word[1)]
Tx PDO1	6041h Status word[1)]

Figure 14 PDO mapping

4.2.2 Load EDS and configuration

As shown in Figure 15 and 16, the node ID of the ABB has to be configured additionally because an EDS file does not provide parameter values.

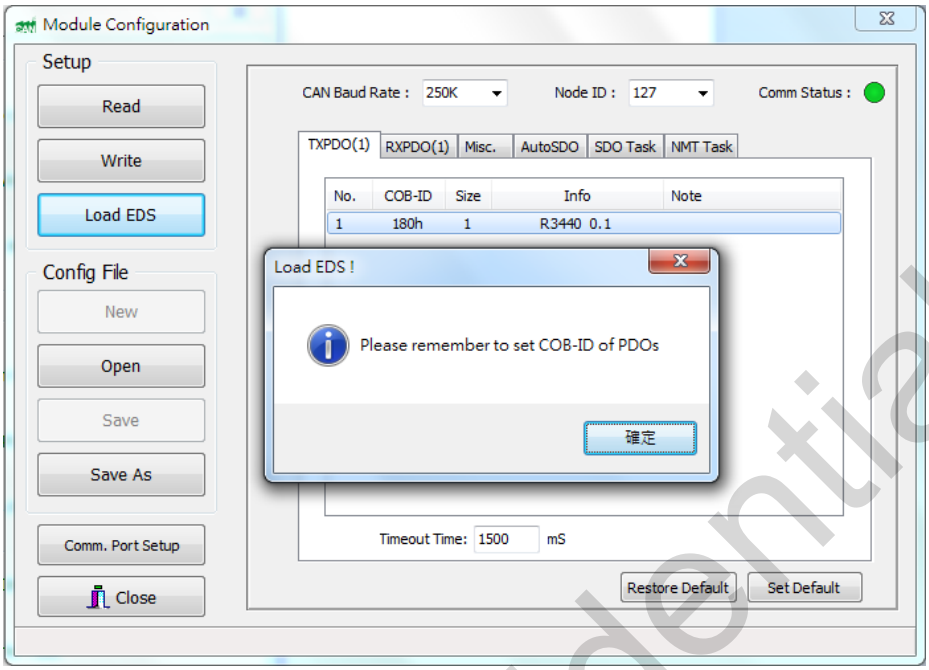


Figure 15 Load EDS

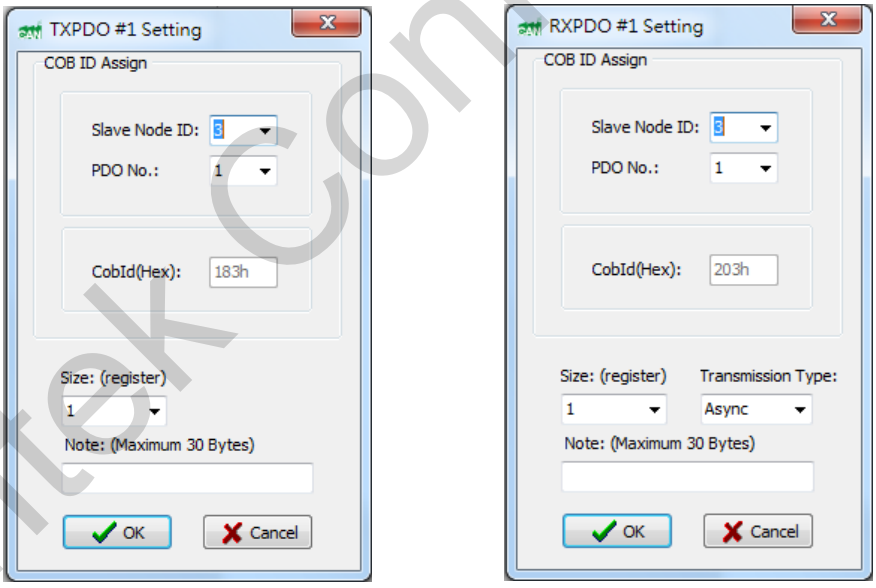


Figure 16 Fill node ID in PDOs

4.2.3 Baud and node ID

4.2.4 Misc setting

The same as defined in the EDS file.

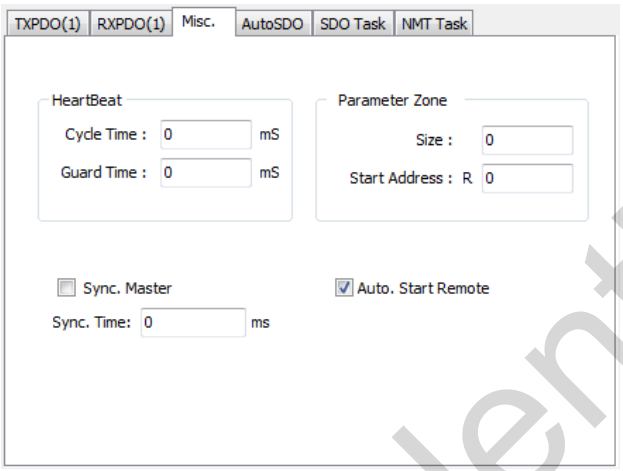


Figure 17 Misc setting after loading EDS

4.2.5 AutoSDO

Plan the AutoSDO operations in accordance with the information as shown in the Figure 18.

Control word:

- Reset the fieldbus communication fault (if active).
- 47Eh (1150 decimal) –> READY TO SWITCH ON
- 47Fh (1151 decimal) –> OPERATING (Speed mode)

or

C7Fh (3199 decimal) –> OPERATING (Torque mode)

Figure 18 ABB Control Word

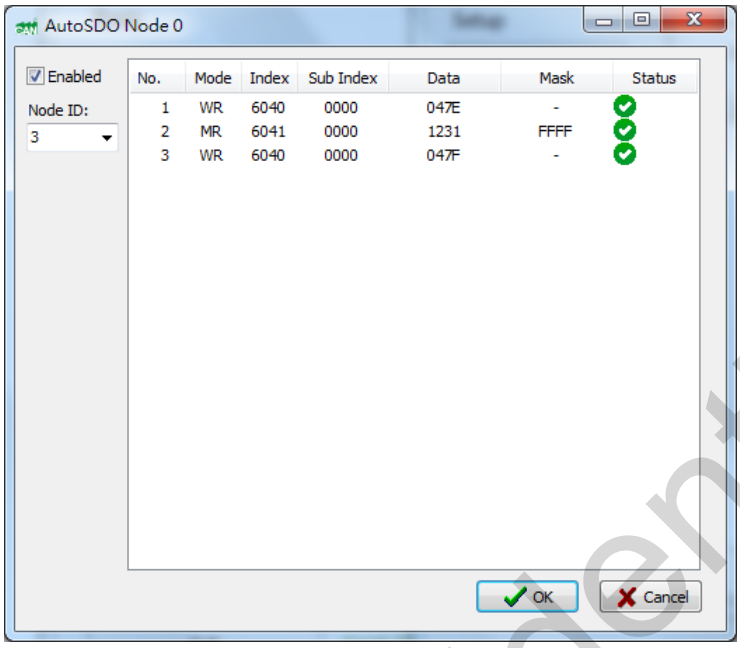


Figure 19 Result of the AutoSDO configuration

The execution sequence expanded from Figure 19:

1) Write 0x47E to 0x6040:00 of the ABB with node ID as 3 in order to put the device in the READY TO SWITCH ON state.
2) Monitor the value of 0x6041:00 of the ABB by comparing it with LOGICAL_AND(0x1231, 0xFFFF)
3) When 0x6041:00 has the correct value, write 0x47F to the ABB to put the device in the OPERATING state.

4.2.6 Configuration complete

Use Write button to save the result of the configuration to the CBCANH.

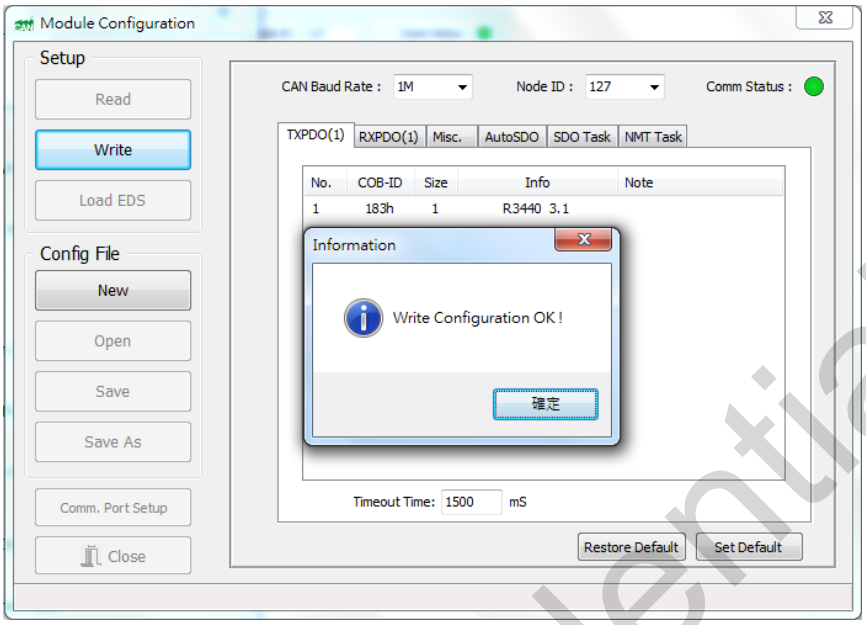


Figure 20 Write back the configuration to the CBCANH

4.2.7 Example ladder program

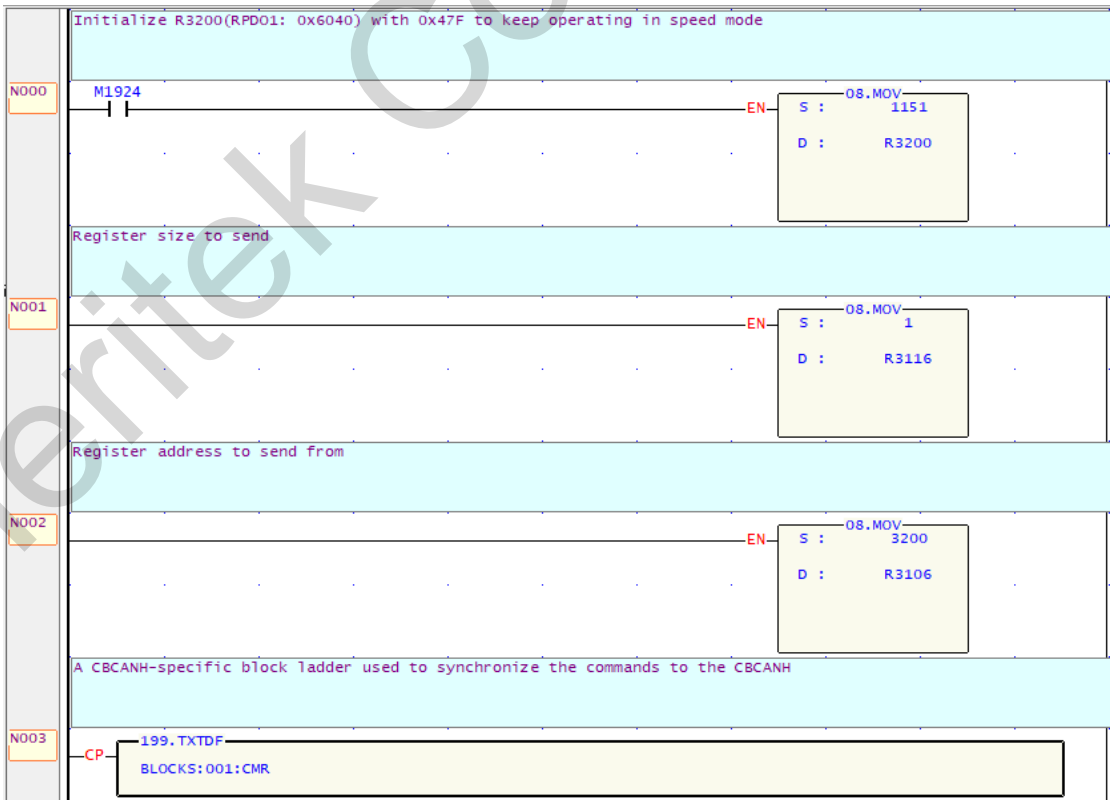


Figure 21 Example program for turning on a motor

4.2.8 Operation steps

1) Configure the CBCANH and the ABB
2) Run the PLC program
3) AutoSDO executes immediately once CBCANH turns on. Therefore, power on ABB first and then the CBCANH. AutoSDO could be redone by triggering block ladder AUTOSDO_CTRL
4) The motor turns on

4.3 Velocity control

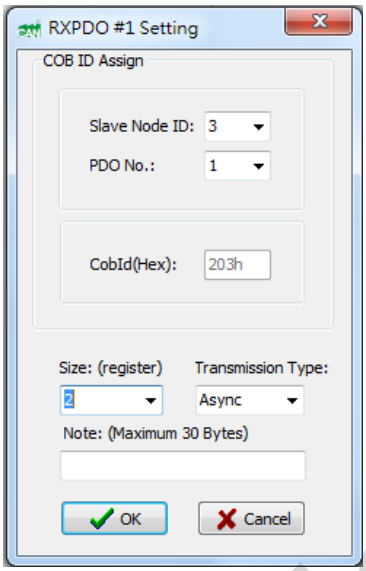
To control velocity, modifying the PDO mapping is necessary. The modified PDO mapping should be the same as Figure shown below. To modify the PDO mapping, follow the following steps.

PDO	Word 1	Word 2
Rx PDO1	6040h Control word[1)]	6042h Target velocity[1)]
Tx PDO1	6041h Status word[1)]	

Figure 22 Modified PDO mapping

4.3.1 Modify PDO configuration

Modify RPDO size from 1 to 2, as shown in the figure below.



4.3.2 Add SDO task

In the object dictionary, index 1600H stores the information on the mappings of first RPDO. Sub-index 0 defines the number of effective mapping of objects, and sub-index 2 represents the second mapped application object. Therefore, Four SDO tasks are created to modify and read data in these two sub-indexes, as shown in Figure 23.

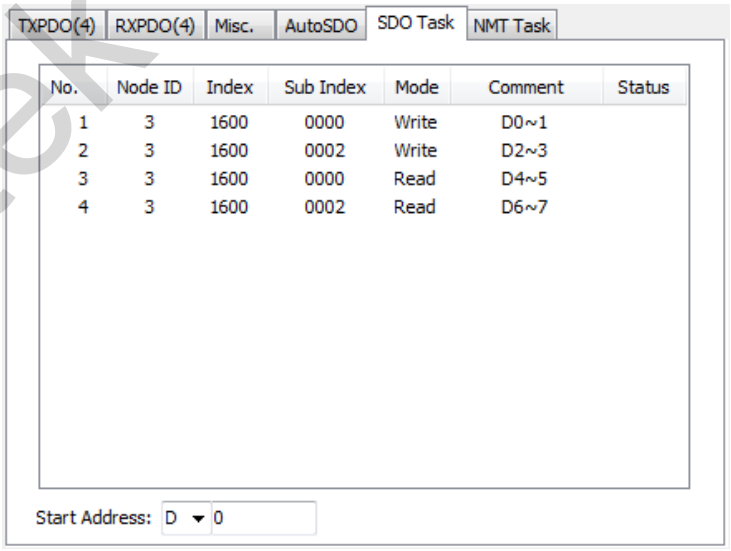


Figure 23 SDO task setup

4.3.3 Add NMT task

The PDO mapping can be modified only if ABB frequency converter is in pre-operational NMT state. Therefore, we need a NMT task to change its NMT state. As shown in Figure 24.

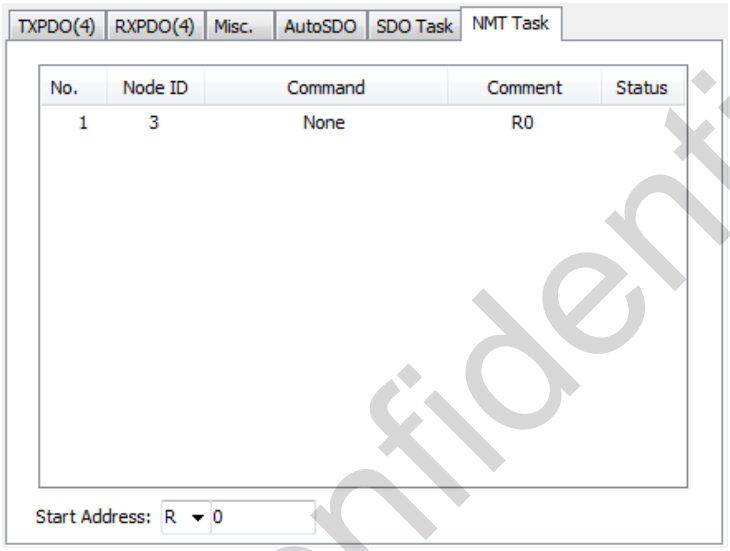


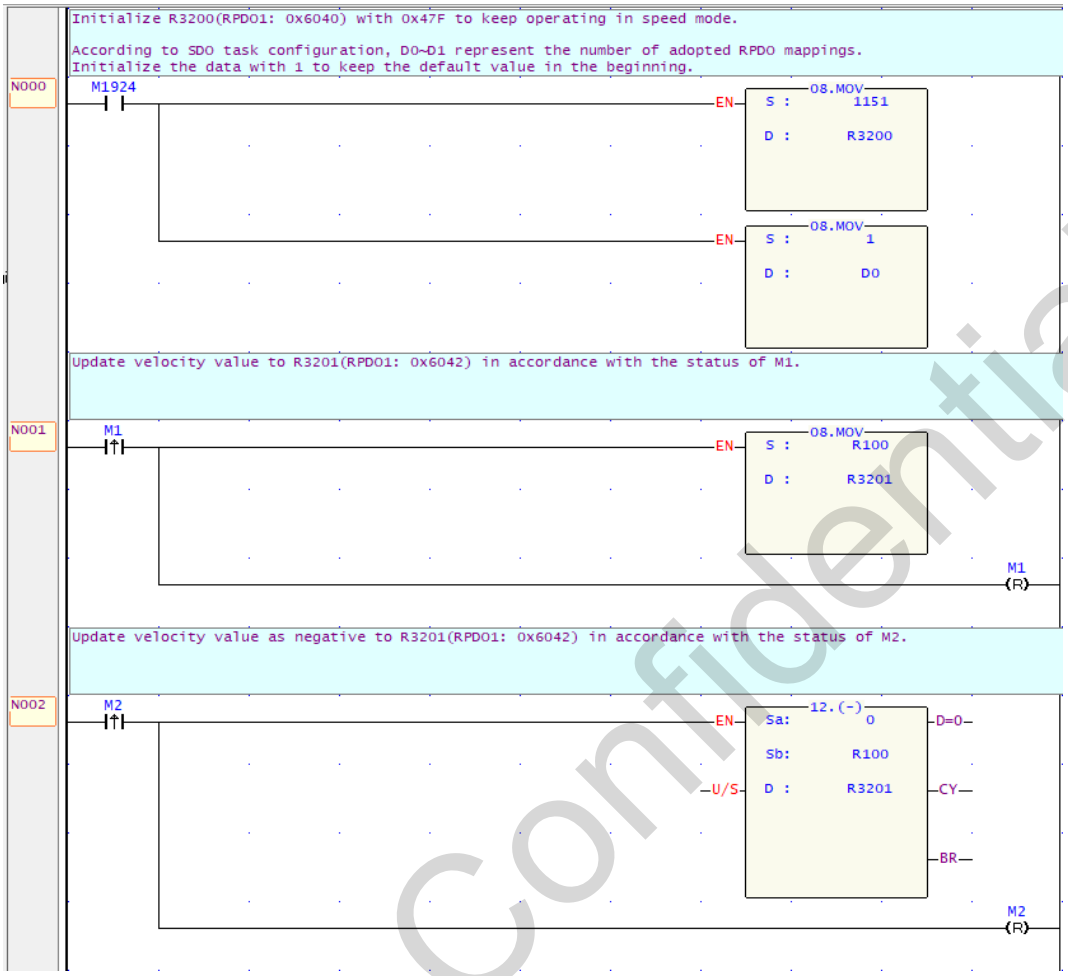
Figure 24 NMT task setup

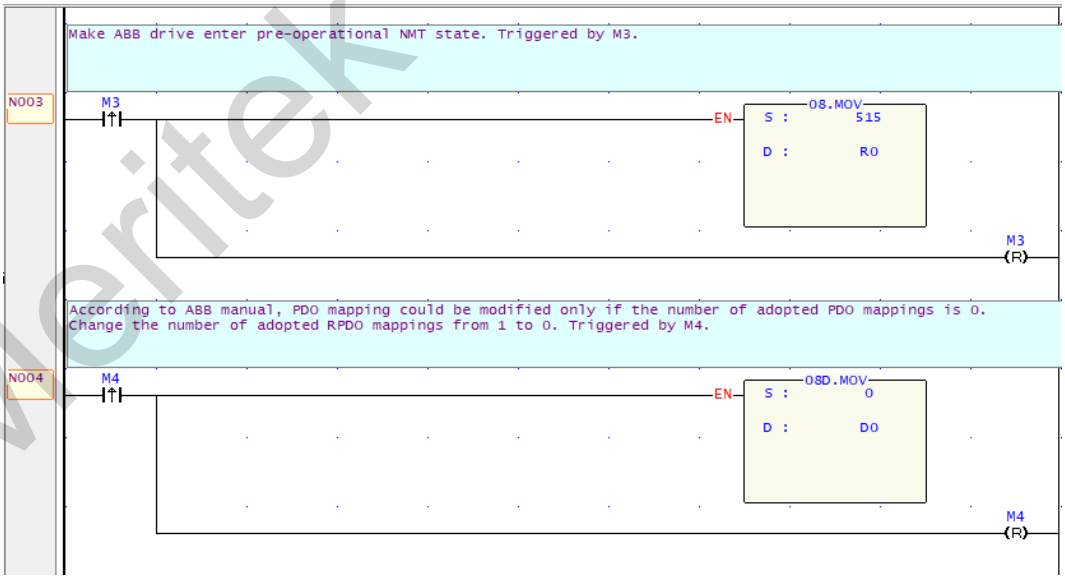
4.3.4 Configuration complete

Use Write button to save the result of the configuration to the CBCANH.

4.3.5 Example ladder program

Trigger M3 ~ M7 sequentially to complete RPDO mapping modification. If ABB drive shows error after RPDO mapping modification, clear the error and redo AutoSDO. After finishing the process, velocity value could be updated through PDO, which is by changing value in R3201.





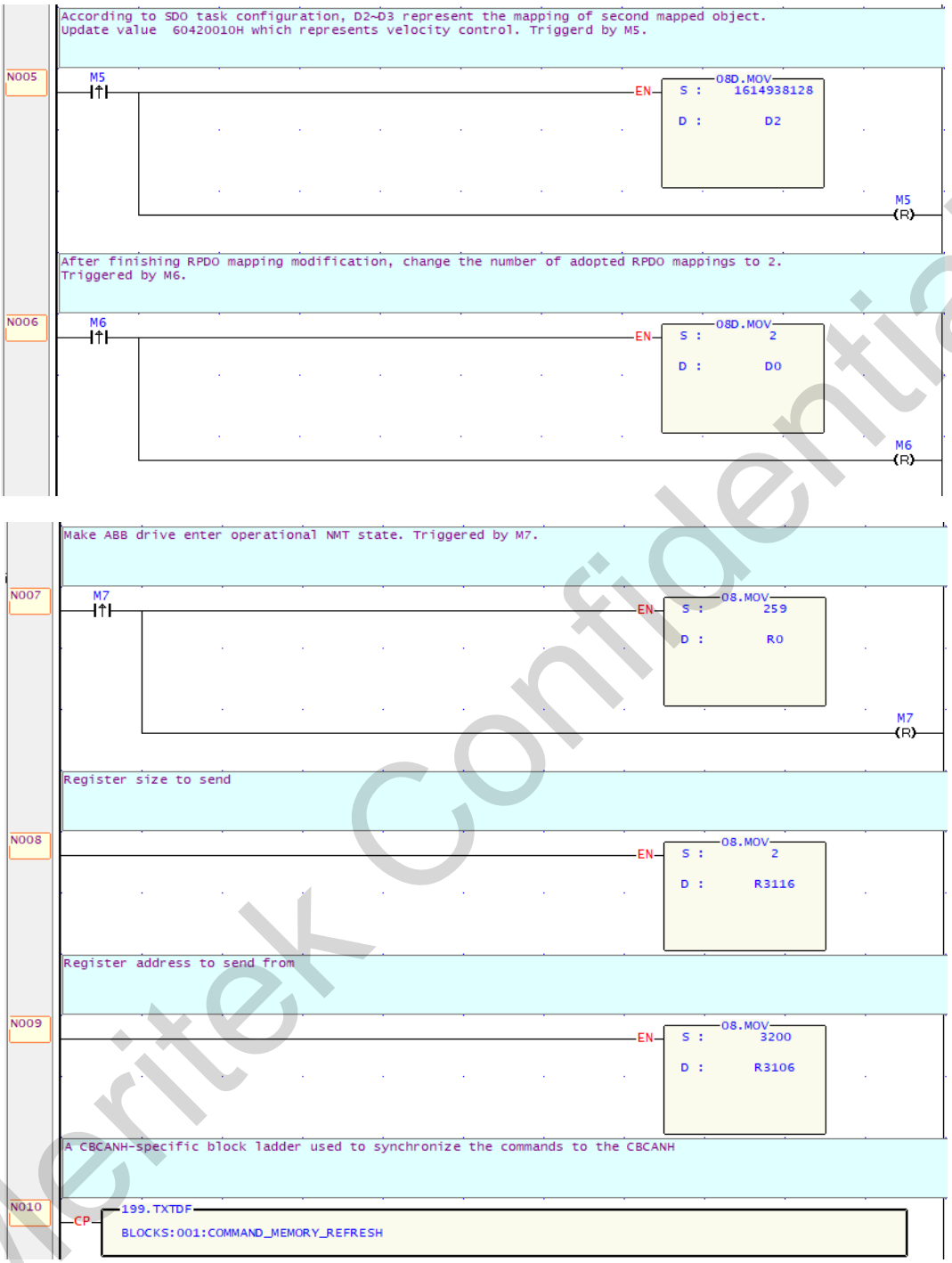


Figure 25 Example program for turning on and controlling velocity of a motor

4.3.6 Operation steps

1) Configure the CBCANH and the ABB

2) Run the PLC program
3) Power on ABB first and then the CBCANH. AutoSDO could be redone by triggering block ladder AUTOSDO_CTRL
4) The motor turns on
5) Control velocity and observe