

ECM-XFU ITE

User Guide

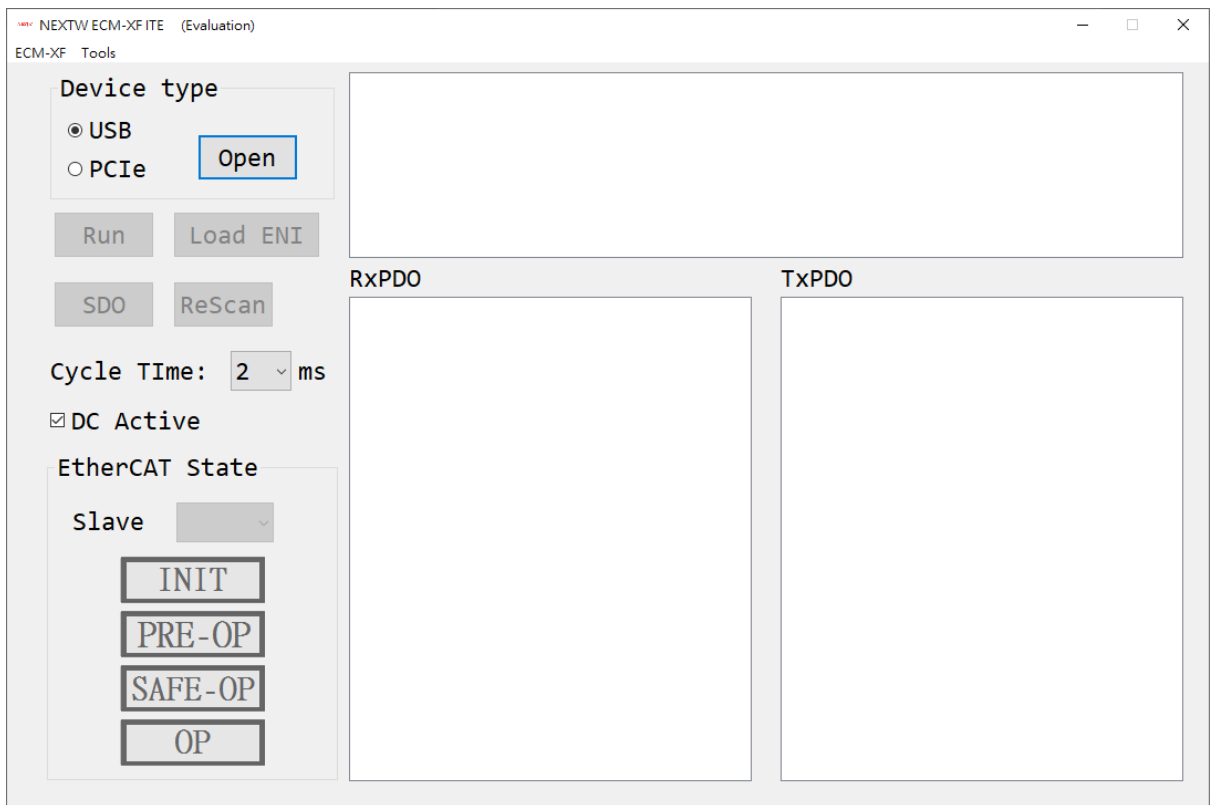
Ver.02
08/31/2021

Preparation

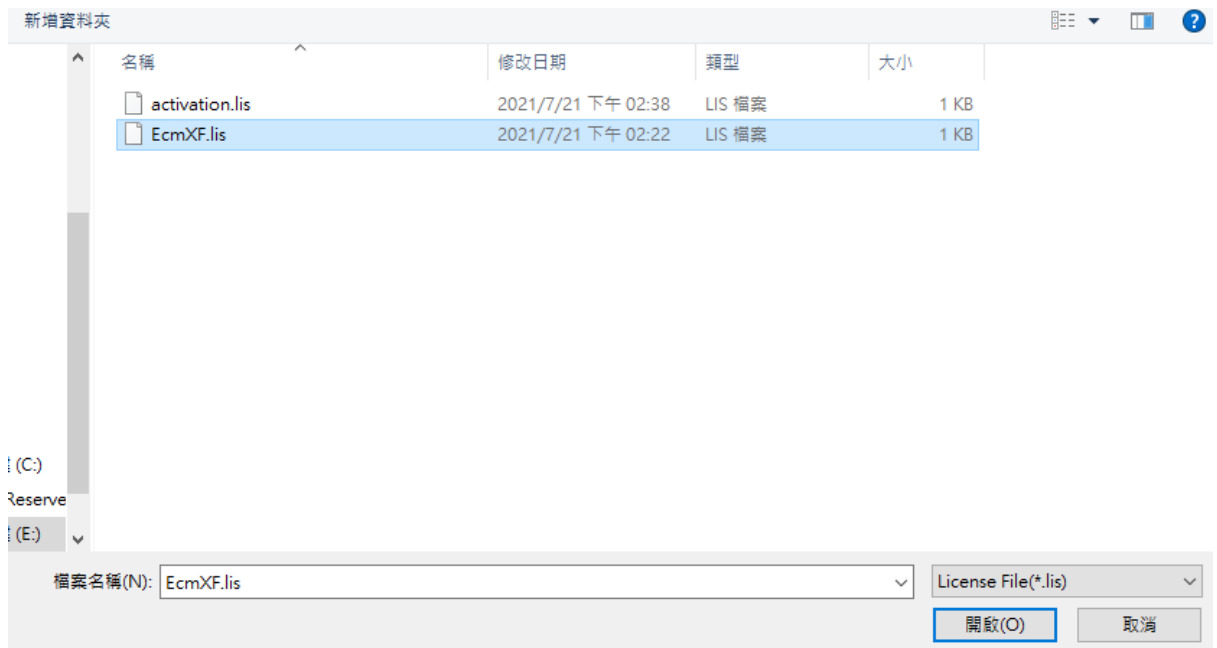
1. Before using ITE, check the ECM-XF board version is up to 1.2 to support USB connection or PCIe connection.
2. Download EcmITE from NEXTW official website. Then Unzip the files, find the EcmITE execution file and execute.
3. Connect your master board to the slave(s).

Version(Evaluation/Licensed)

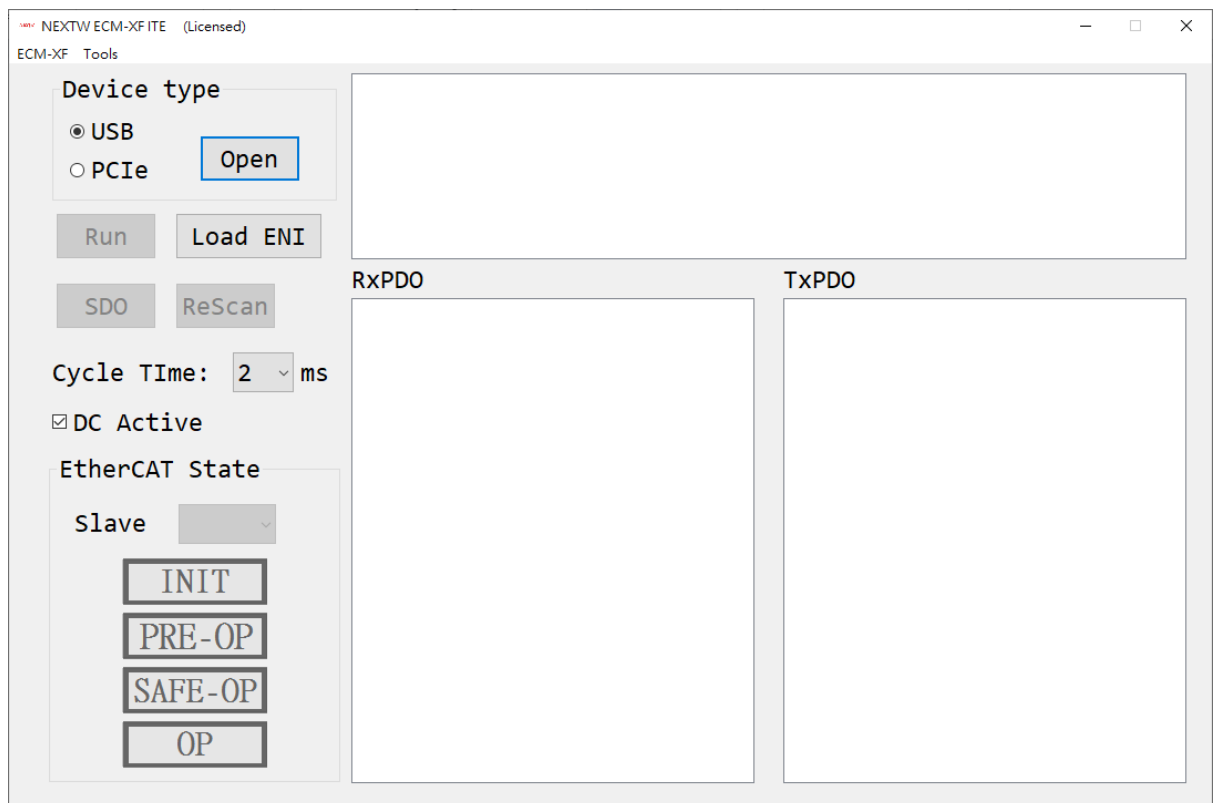
The interface is shown below. The download version would be the evaluation version shown in the upper left window bar. It includes fundamental functions for reading and controlling slave default PDO. The licensed version can be upgraded by sending the “activation.lis” file to service@nextw.com.tw.



Once having the license, put the license file “EcmXF.lis” into the same folder of the ITE. Then click the “ECM-XF” on the toolbar and select “License”. Choose the license file and click open.



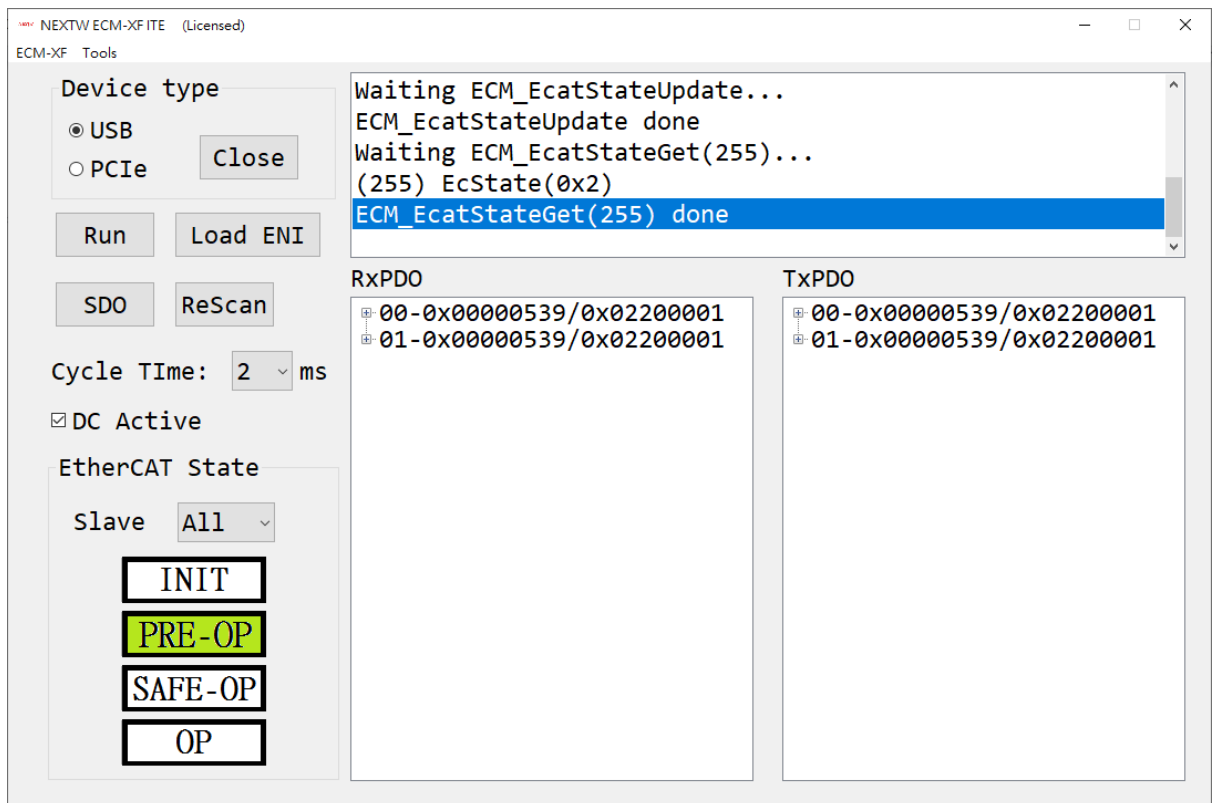
The upper left window would show the “Licensed” to unlock advanced functions.



Functions

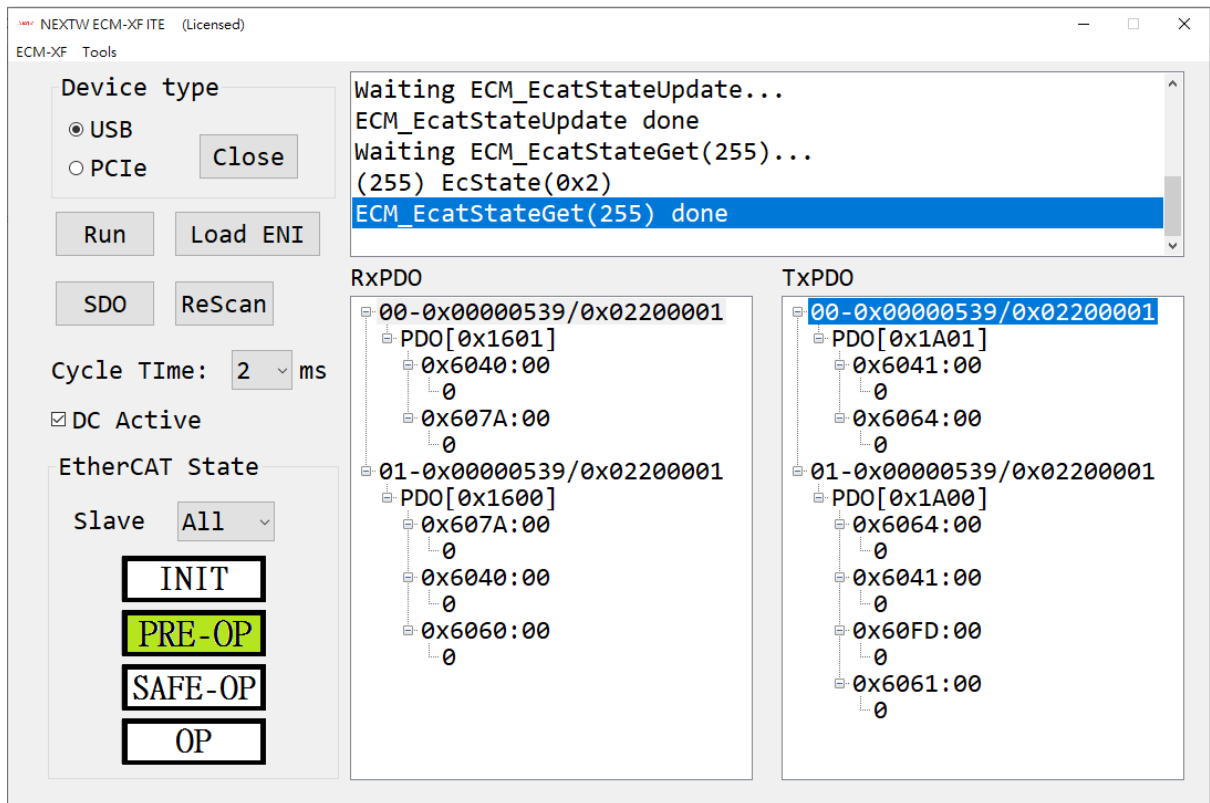
Common Use Sequence

The ITE supports USB and PCIe connection. Choose one for your application and push the button “Open”.



After opening the connection, the state should be in the Pre-OP state. The upper right text box will show the firmware version of the ECM-XF chip. The default setting of the slave will be presented at the RxPDO and TxPDO text boxes.

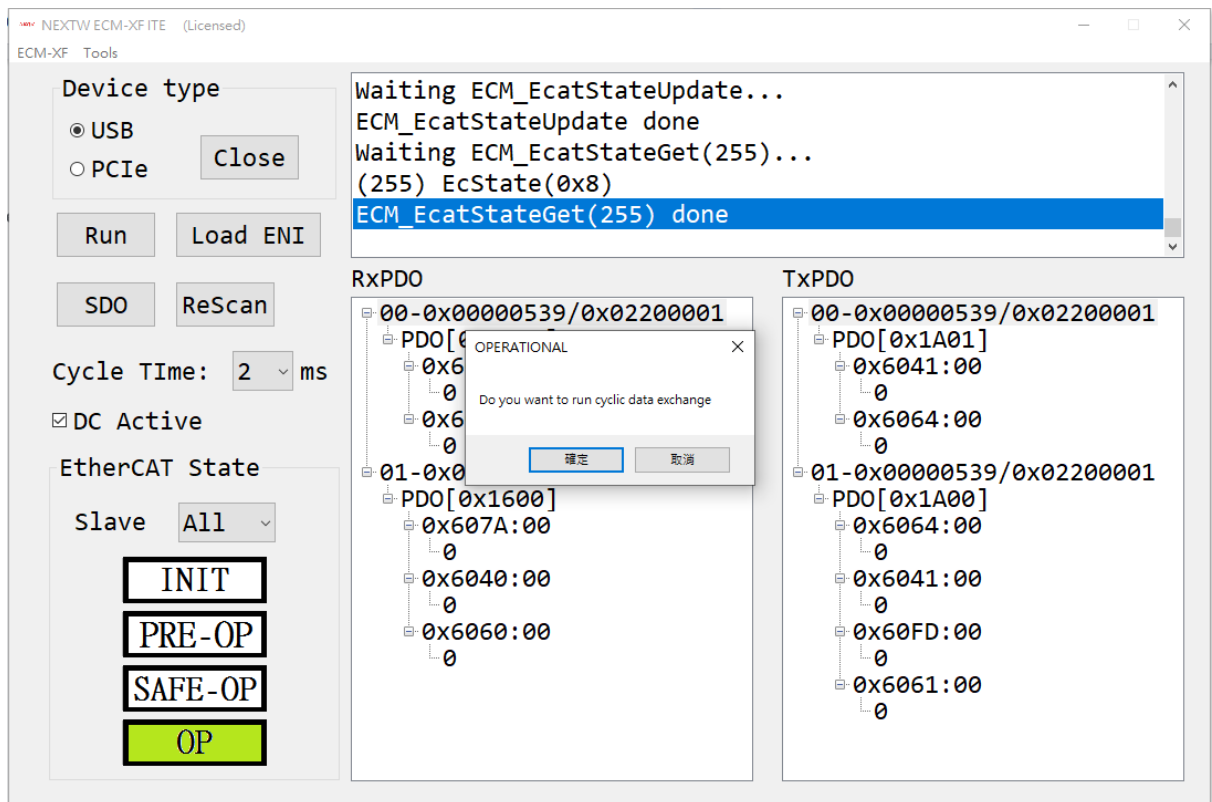
By clicking the “+” symbols in the front of the contents in RxPDO and TxPDO text boxes, the details will be presented. The picture below shows the YASKAWA drive default setting at CSP modes.



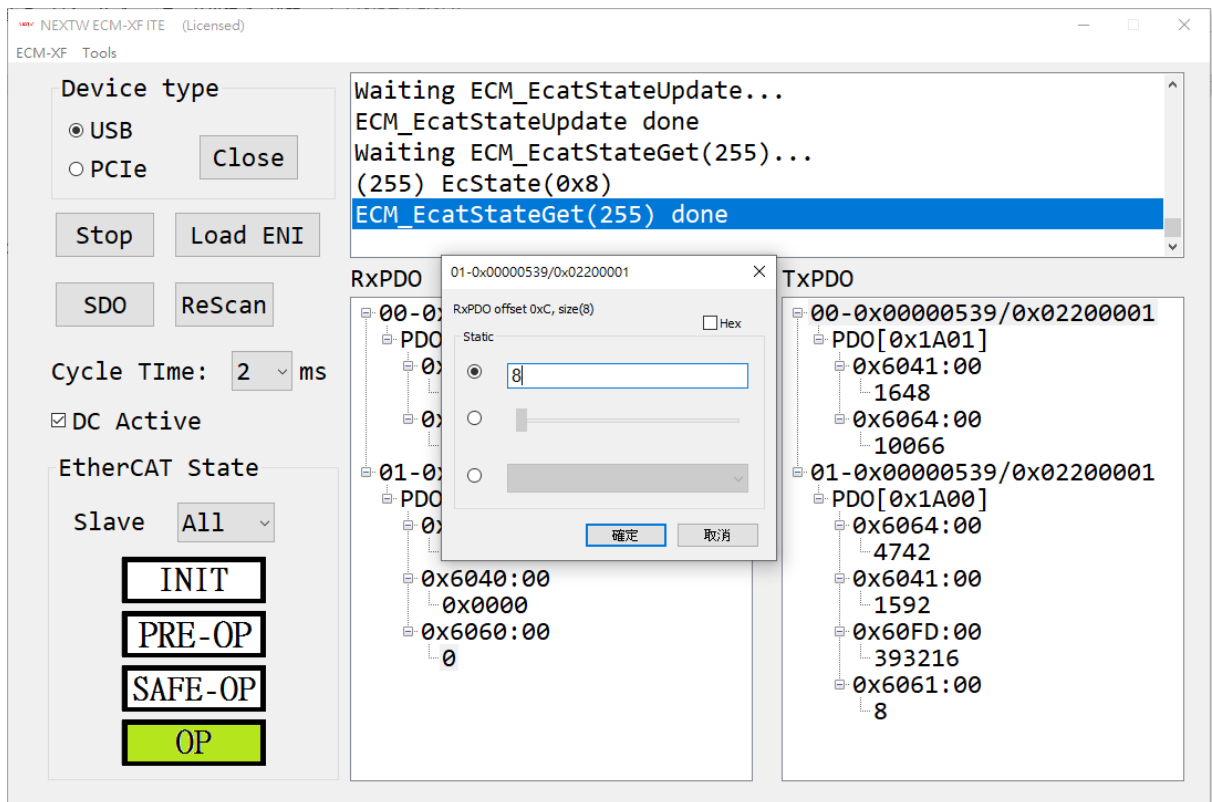
In the Pre-OP state, you can read/write the SDO, load the ENI, use SII editor and code generator in tools.

Next change state into Safe-OP then into OP state. Beware the state needs to follow the sequence from top to bottom, otherwise the switching state would fail.

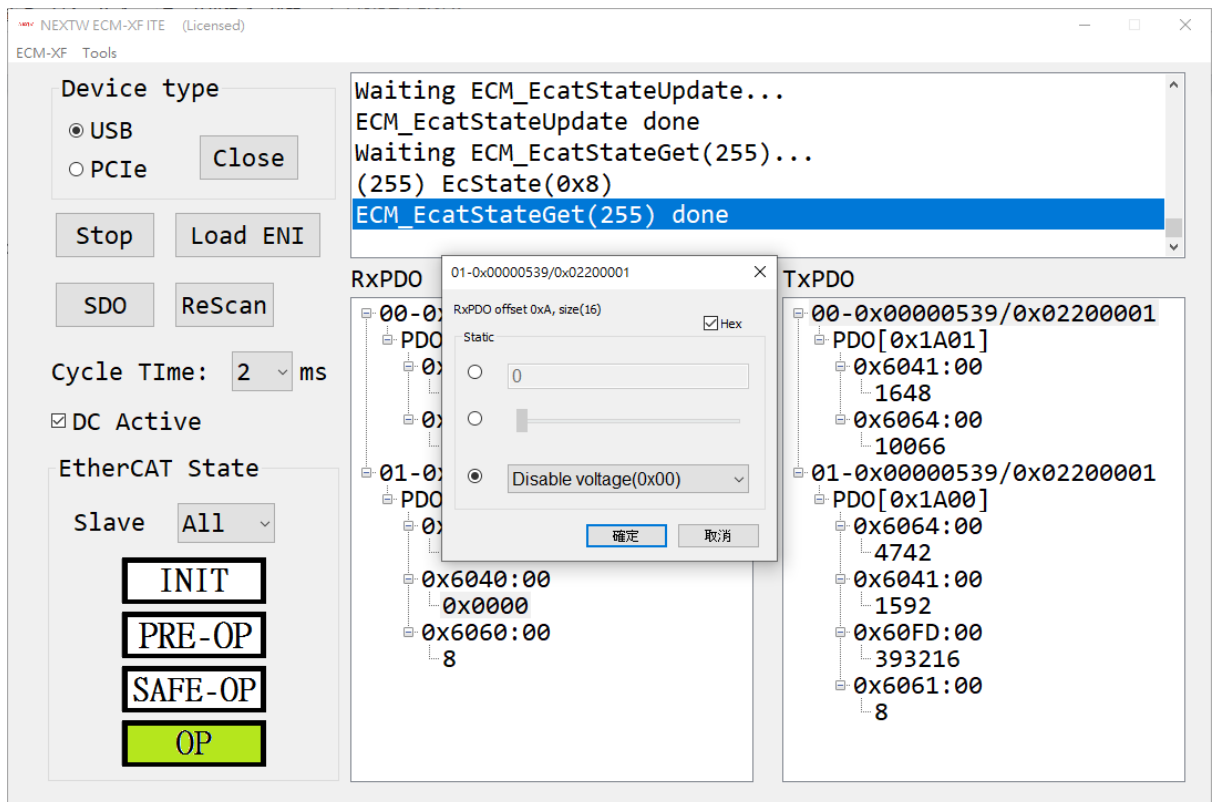
When pushing to the OP state, the pop out window will show the confirmation of the cyclic data exchange. It has the same function as the "Run" button.



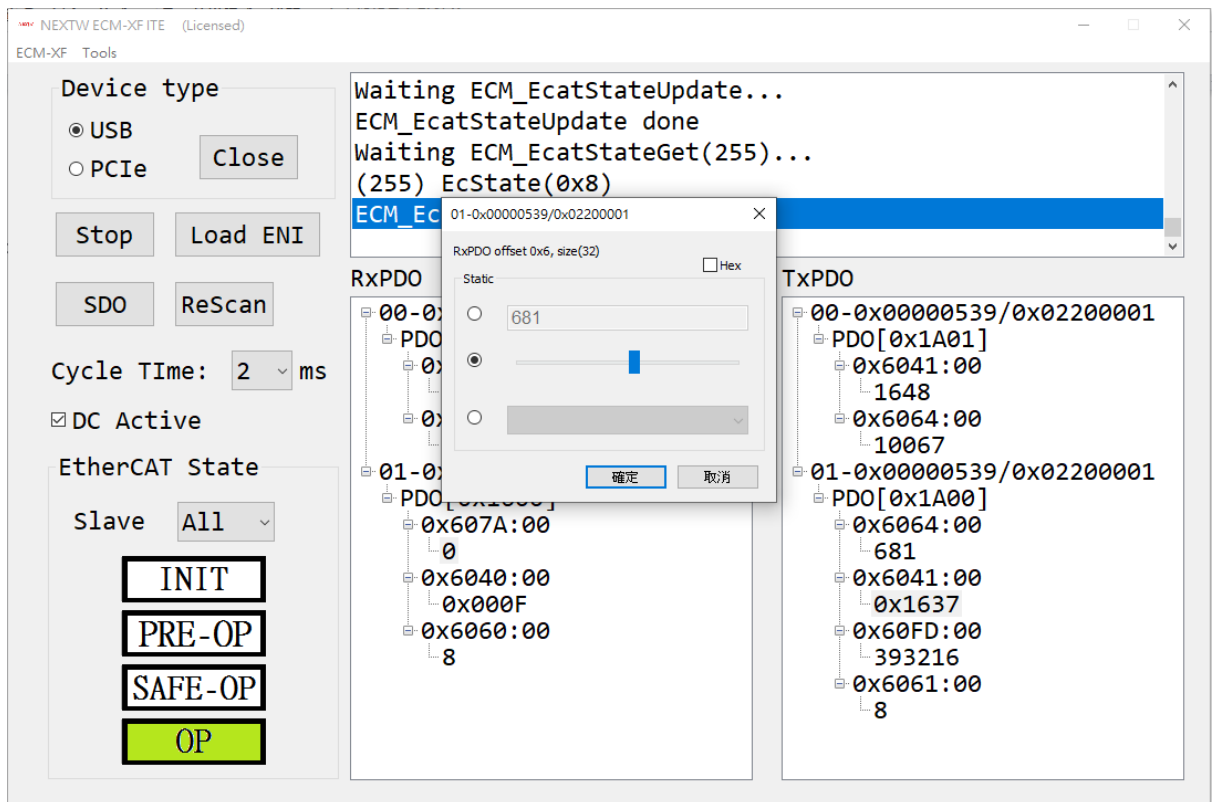
In the run mode, the data can be changed into your application. The TxPDO shows the YASKAWA current status. The data can be right clicked to switch into hexadecimal if you need. Double clicking the value section can change the value.



Double clicking the 0x6040 to change the 402 state machine in the run mode. There is no need to press OK everytime to change the state. Observing the TxPDO corresponding to 0x6041 will be changed when you click the different control word.

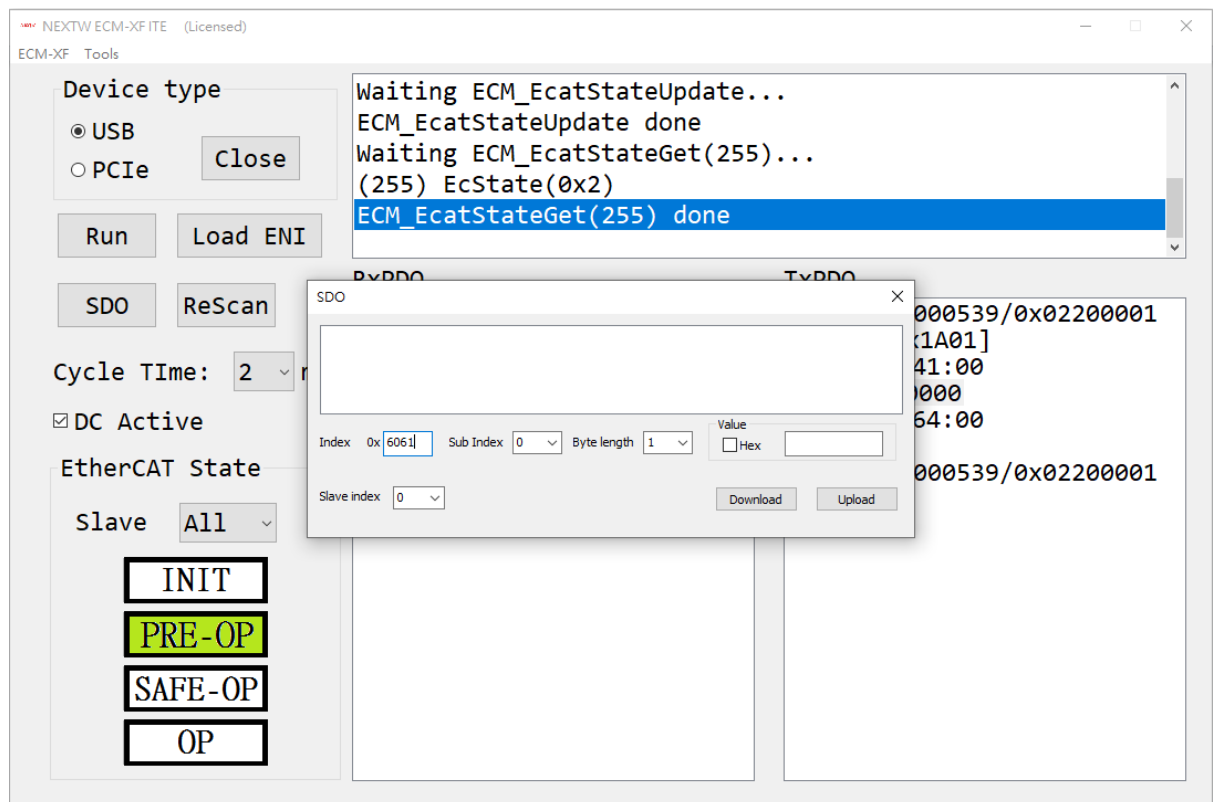


By choosing "Enable Operation(0xF)" to servo on the driver, then you will see the value in 0x607A, RxPDO is equal to the value of 0x6064, TxPDO. Now double clicking the value below the 0x607A to set a new position. You can also use the bar to test the motor. The value of target position(0x607A) will be synchronized with the actual position(0x6064) when you are in the run mode.

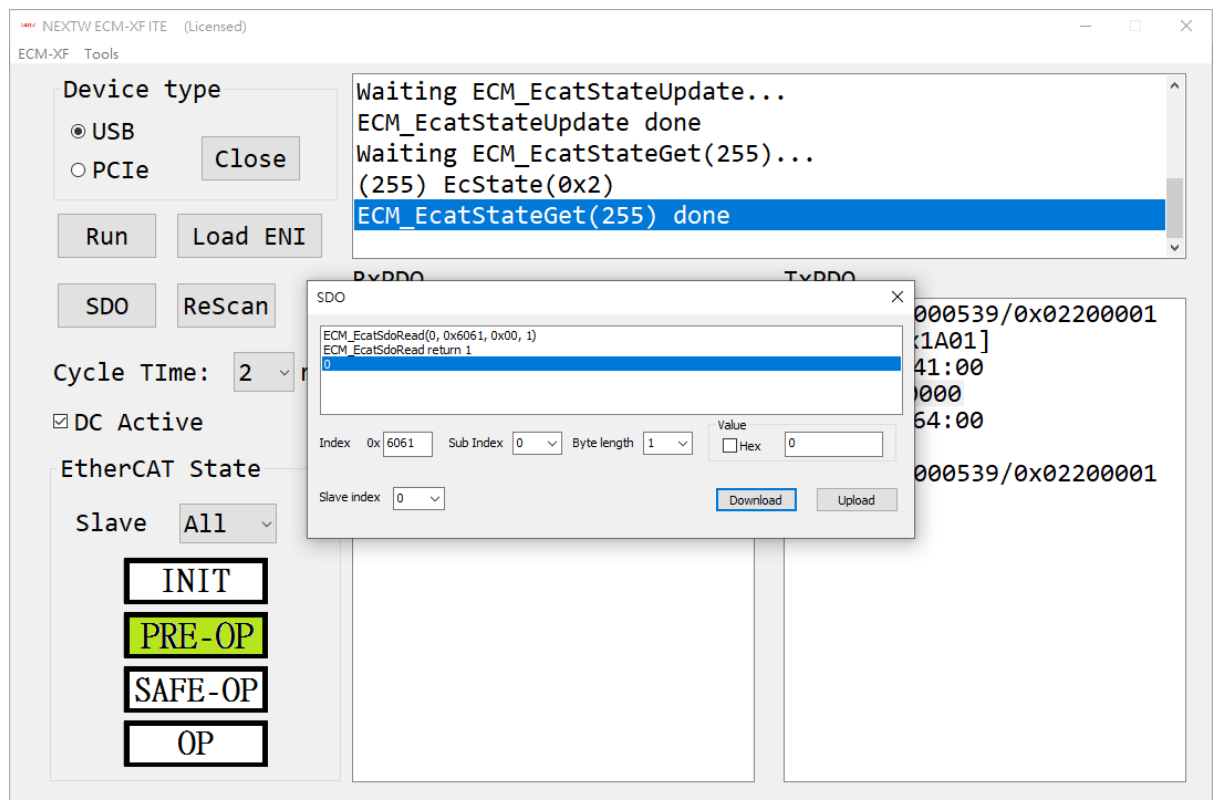


SDO

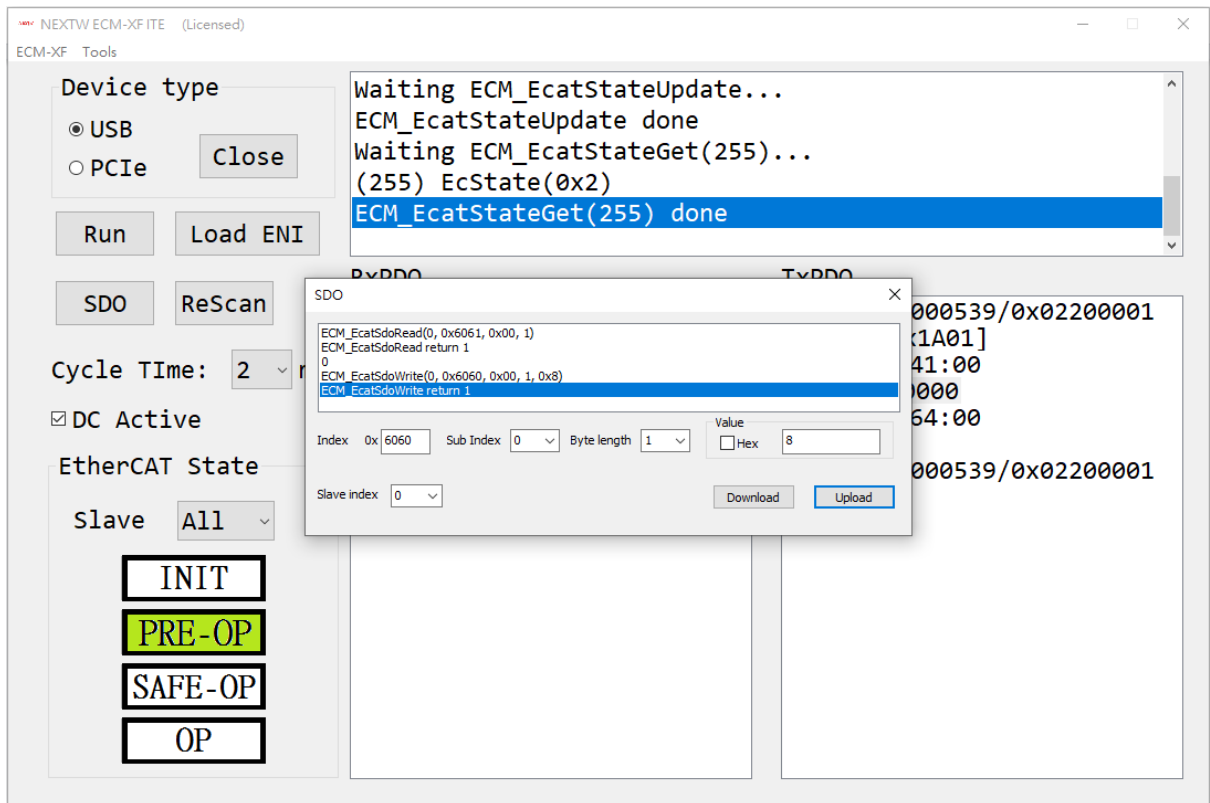
SDO data exchange can be used from pressing the SDO button. For example, Index 6061 displays the operation mode, then press download to get the current operation mode.



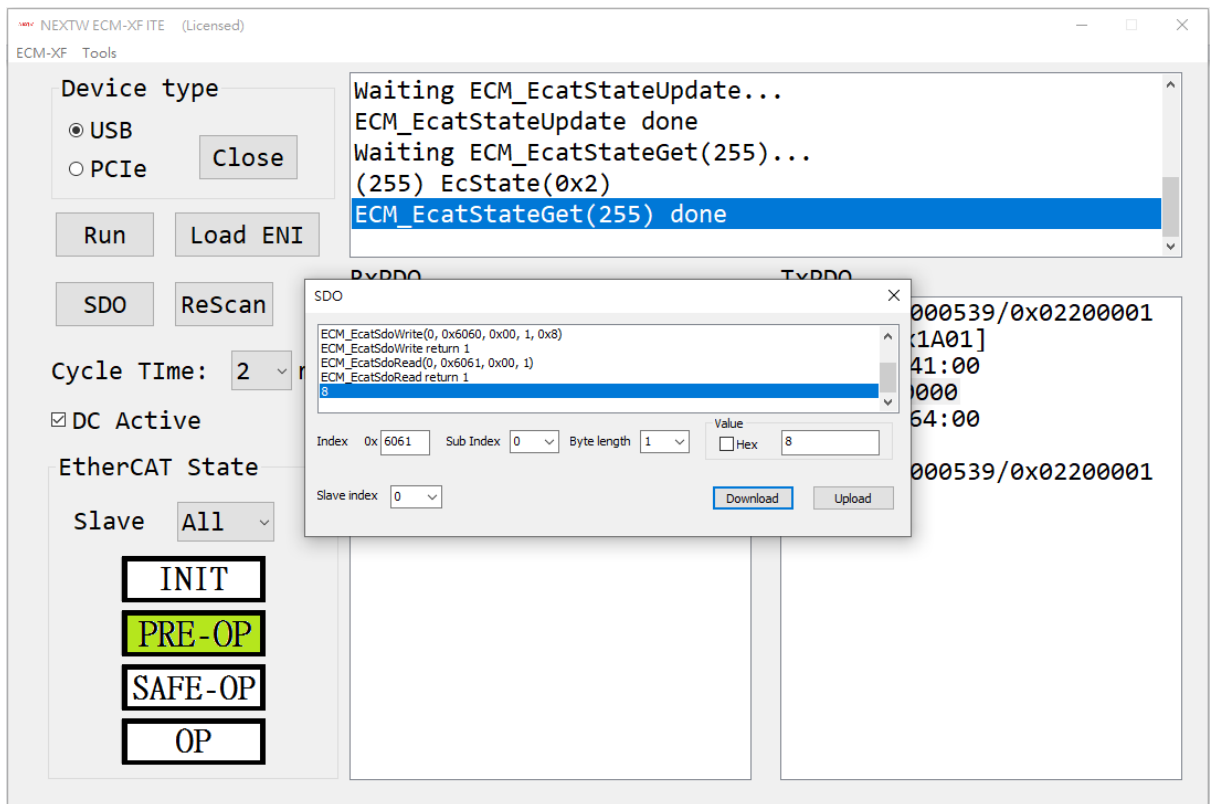
The result shows the operation mode is 0.



Then use index 0x6060 set operation mode with CSP mode (0x8) and upload the setting to EtherCAT master.

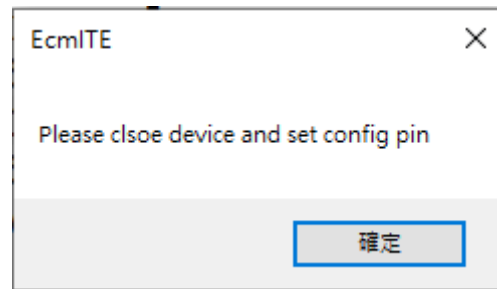


After uploading, repeat 0x6061 to display current operation mode, it shows the operation mode has been changed into 8 for CSP.

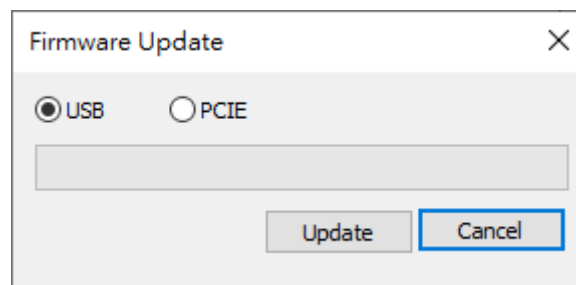




Update firmware

The ITE supports ECM-XF firmware update. By switching the board into configuration mode and clicking the toolbar “ECM-XF” then select “FW update”. The configuration mode please see the board manual or contact the manufacturer. If your board is not in the configuration mode, you would see the window like in the picture below.



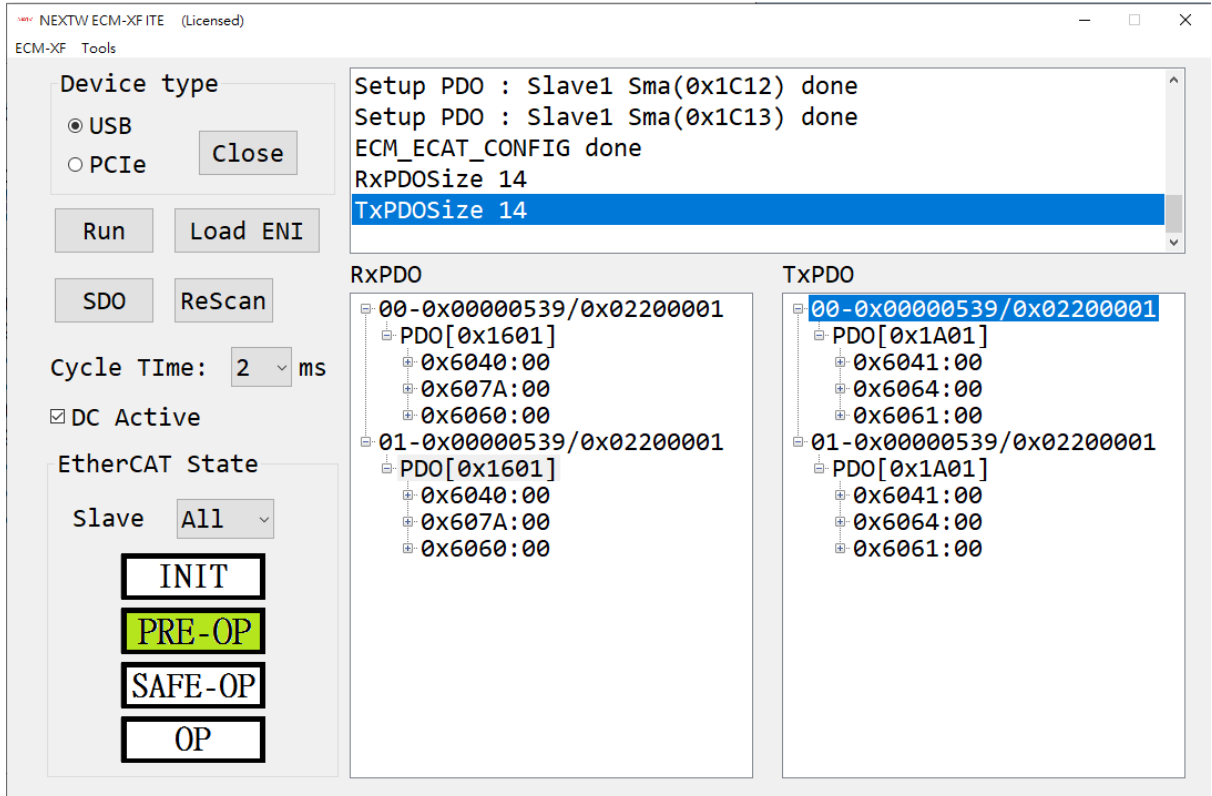
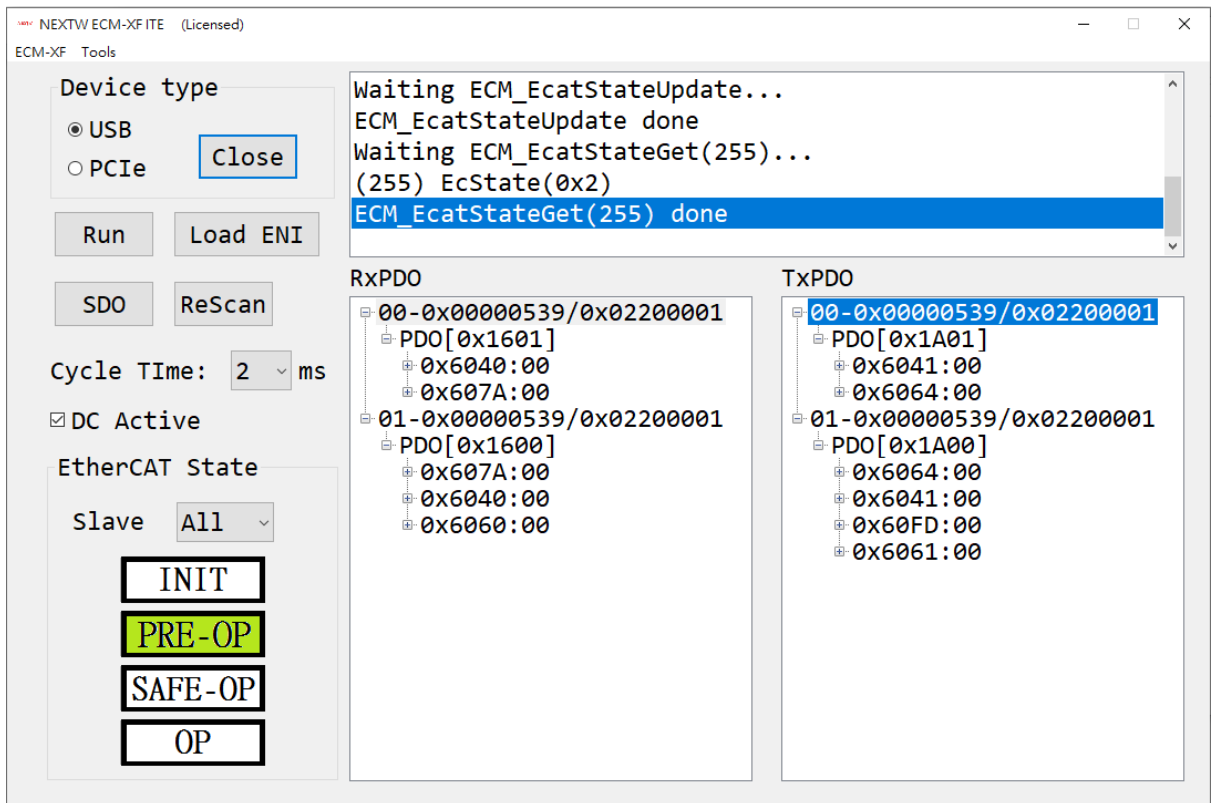
If you are in the configuration mode, the window will show to select the USB or PCIe device for update. Select the corresponding binary file to update the firmware.



 xf16encrypt.bin
 xfu16encrypt.bin

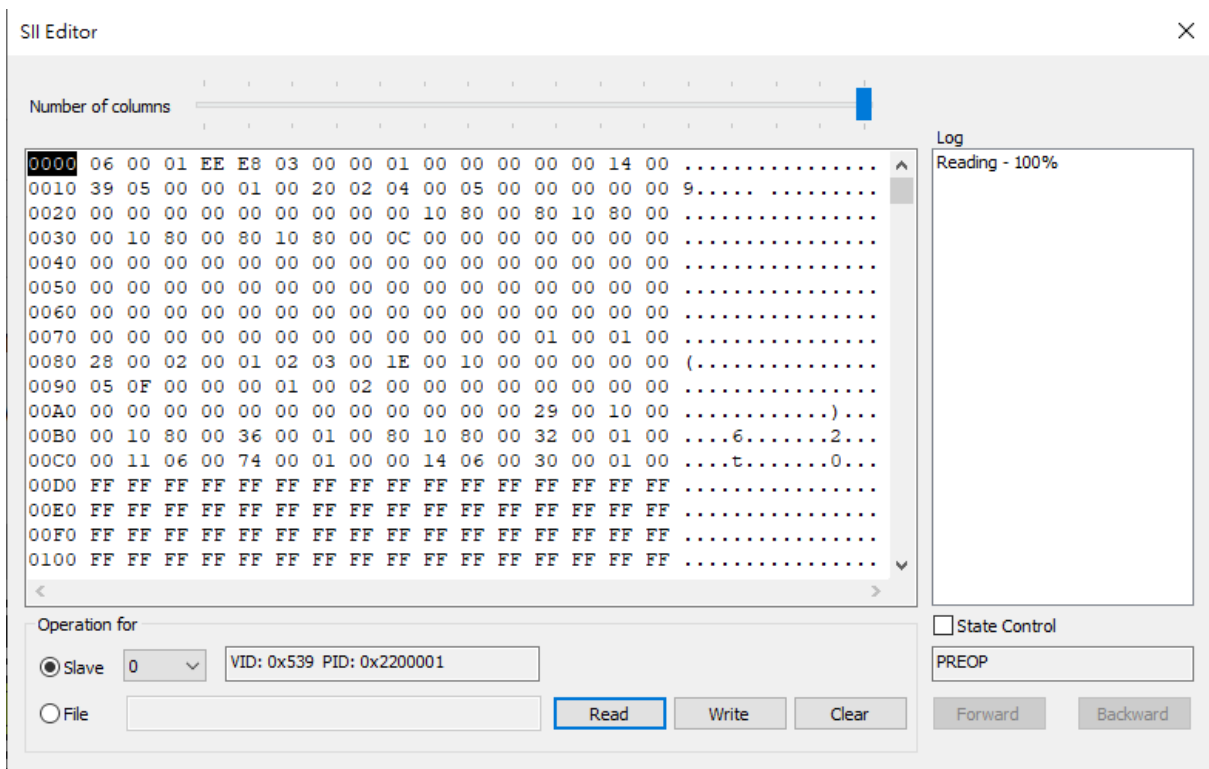
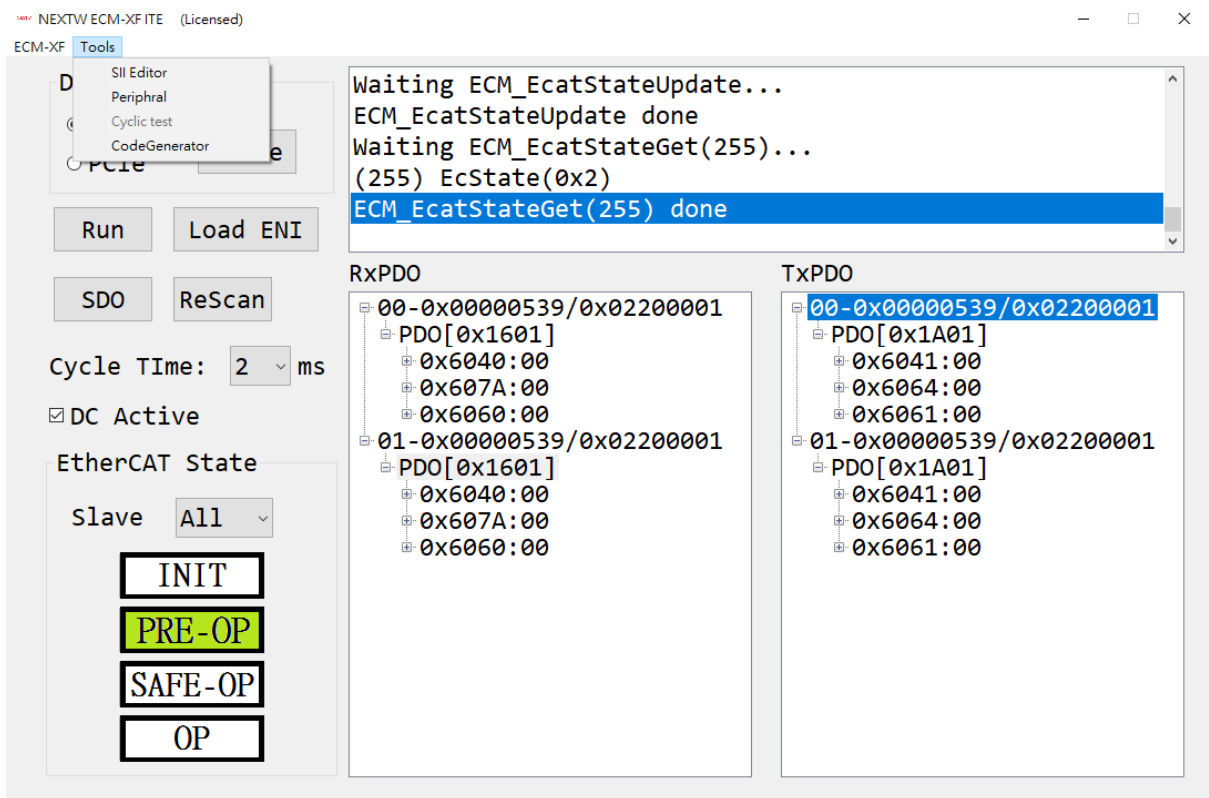
Load ENI

The ITE supports configured PDO ENI for fast application. Open the device and press the Load ENI button to choose the ENI file to change the slave default PDO structure. The picture below shows the RxPDO and TxPDO changes are configured by the ENI settings.



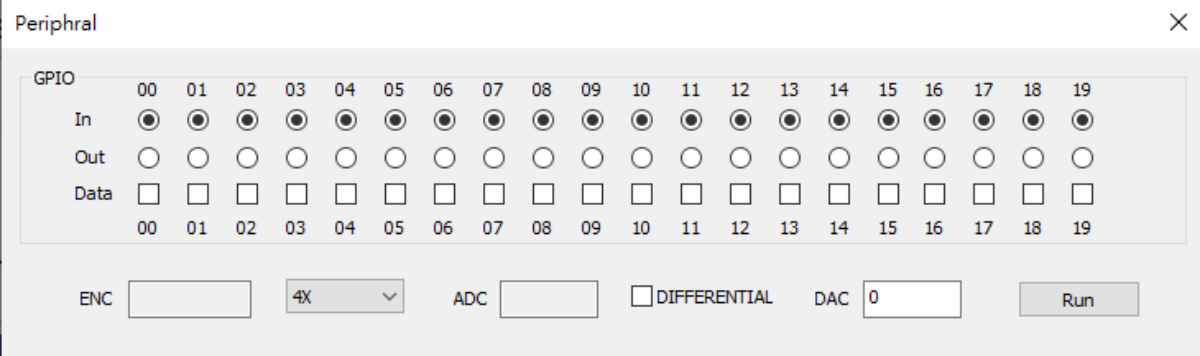
SII editor

The toolbar has the SII editor for convenient reading the slaves SII and saving the files. Use this editor in the Pre-OP state is recommended.



Peripheral

The peripheral window is for ECM-XFU GPIO/QEI/ADC function.



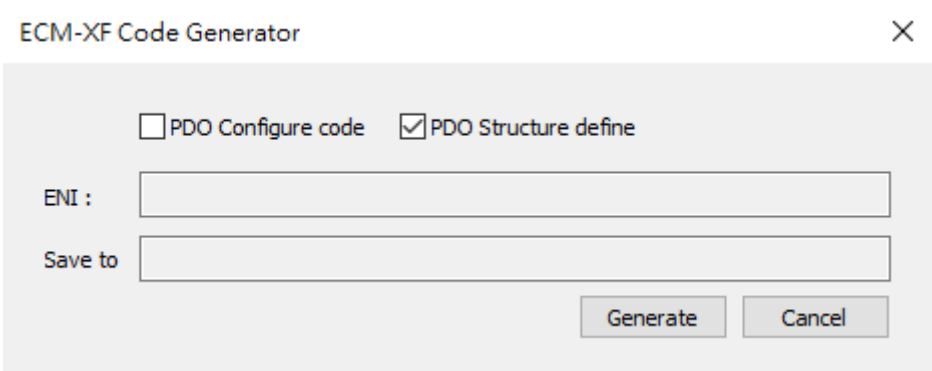
The Peripheral window shows a table for configuring GPIO pins (00 to 19). The table has three rows: In, Out, and Data. The In row has all pins selected with radio buttons. The Out and Data rows have all pins unselected with checkboxes. Below the table, there are fields for ENC, a dropdown menu set to 4X, ADC, a checkbox for DIFFERENTIAL, DAC set to 0, and a Run button.

| GPIO | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| In | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="radio"/> |
| Out | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Data | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

ENC 4X ADC ☐ DIFFERENTIAL DAC

Code Generator

The code generator is for generating the configuration .c file from ENI and the PDO structure .h file from the current configuration. It is important to make sure loading the ENI first to configure the PDO structure before generating the .h file.



The ECM-XF Code Generator window has two checkboxes: PDO Configure code (unchecked) and PDO Structure define (checked). Below these are two text input fields labeled ENI : and Save to. At the bottom right are Generate and Cancel buttons.

☐ PDO Configure code ☒ PDO Structure define

ENI :

Save to

The configure code(.c file) and PDO structure define(.h file) files can be put into your application code.

The easy import way is to copy c file function

ECM_ConfigPDO_CodeGen(void) to the main.c as function like the picture shows below.

```

#define ECM_TBL_CNT_CC 4

int ECM_ConfigPDO_CodeCen(void)
{
    int nDone = 0;
    int nRet = 0;
    PDO_CONFIG_HEAD stEcmPdoTbl[ECM_TBL_CNT_CC] = {
        {0x539,0x2200001,0,1,0x1C12,{0x1601,0x0,0x0,0x0},{3,0,0,0},{16,0,0x6040},{32,0,0x607A},{8,0,0x6060}},
        {0x539,0x2200001,0,1,0x1C13,{0x1A01,0x0,0x0,0x0},{3,0,0,0},{16,0,0x6041},{32,0,0x6064},{8,0,0x6061}},
        {0x539,0x2200001,1,1,0x1C12,{0x1601,0x0,0x0,0x0},{3,0,0,0},{16,0,0x6040},{32,0,0x607A},{8,0,0x6060}},
        {0x539,0x2200001,1,1,0x1C13,{0x1A01,0x0,0x0,0x0},{3,0,0,0},{16,0,0x6041},{32,0,0x6064},{8,0,0x6061}} };
    for (int i = 0; i < ECM_TBL_CNT_CC; i++) {
        if (stEcmPdoTbl[i].PDOCnt) {
            nRet = ECM_EcatSdoSetPdoConfig(&stEcmPdoTbl[i]);
            if (nRet == 1) {
                nDone++;
            }
        }
    }
    return nDone;
}

int main()
{
    uint16_t ul6SpiDataSize = 992; //fixed in USB
    uint8_t u8Version;

    PDO_CONFIG_HEAD RxPDOConfig[2];
    PDO_CONFIG_HEAD TxPDOConfig[2];
    uint8_t RxData[TEST_SPI_DATA_SIZE] = { 0 };
    uint8_t TxData[TEST_SPI_DATA_SIZE] = { 0 };

```

And in the configuration part just call the function
“ECM_ConfigPDO_CodeGen(void)”

```

nret = ECM_StateCheck(0xFF, EC_STATE_PRE_OP, 10000); // Set mode must be at PRE-OP state

if (nret == 0) {
    return -1;
}

std::cout << "All Slaves are in PRE-OP state\n";
std::cout << "Assign and configure PDO\n";

ECM_ConfigPDO_CodeCen();

// Call ECM_EcatReconfig() after setting PDO configures
nret = ECM_EcatReconfig();
if (nret < 0) {
    std::cout << "ECM_ECACONFIG error : " << nret << "\n";
    return -1;
}

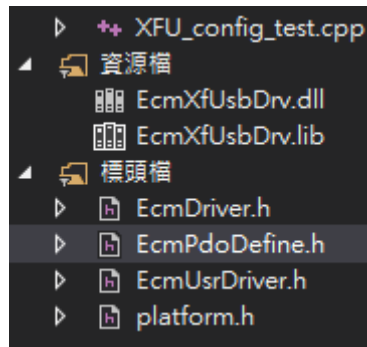
```

The EcmPdoDefine.h is the same functional file as PdoDefine.h in SPI version. Use #include EcmPdoDefine.h to import it into the main.c file.

```

#include <EcmUsrDriver.h>
#include <platform.h>
#include <EcmPdoDefine.h>

```

Maintainance Log

| Version | Date | Note |
|---------|------------|--|
| 00 | 07.28.2021 | |
| 01 | 08.18.2021 | Add SDO content |
| 02 | 08.31.2021 | Add Load ENI, SII editor, Peripheral, and Code generator |