Application Note

FESTO

CMMT EtherCat controlled by C++

This document describes how to control the CMMT EtherCat version with the programming language C++, using the SOEM (Simple Open Source EtherCat Master) library.

CMMT-AS
CMMT-ST

Title ..... CMMT EtherCat with C++
Version 1.11
Document no. 100338
Original ..... en
Author ..... Festo
Last saved 10.10.2023

Copyright Notice

Legal Notice

Hardware, software, operating systems and drivers may only be used for the applications described and only in conjunction with components recommended by Festo SE & Co. KG.

Festo SE & Co. KG does not accept any liability for damages arising from the use of any incorrect or incomplete information contained in this documentation or any information missing therefrom.

Defects resulting from the improper handling of devices and modules are excluded from the warranty.

The data and information specified in this document should not be used for the implementation of safety functions relating to the protection of personnel and machinery.

No liability is accepted for claims for damages arising from a failure or functional defect. In other respects, the regulations with regard to liability from the terms and conditions of delivery, payment and use of software of Festo SE & Co. KG, which can be found at www.festo.com and can be supplied on request, shall apply.

All data contained in this document do not represent guaranteed specifications, particularly with regard to functionality, condition or quality, in the legal sense.

The information in this document serves only as basic information for the implementation of a specific, hypothetical application and is in no way intended as a substitute for the operating instructions of the respective manufacturers and the design and testing of the respective application by the user.

The operating instructions for Festo products can be found at www.festo.com/sp .

Users of this document (application note) must verify that all functions described here also work correctly in the application. By reading this document and adhering to the specifications contained therein, users are also solely responsible for their own application.

Table of content

1 Components/Software used

Type/Name	Version Software/Firmware	Date of manufacture
CMMT-AS-C4-3A-MP-S1	17.0.8.48_release	
Visual Studio	2019	
SOEM library (included)	1.4.0	
Npcap	1.77	
Automation Suite	2.6.0.481	

Table 1.1: 1 Components/Software used

1.1 Topology of the tested system

The Network adapter from the EtherCat Master (controlling system), is directly connected to the XF1 IN port on top of the CMMT.

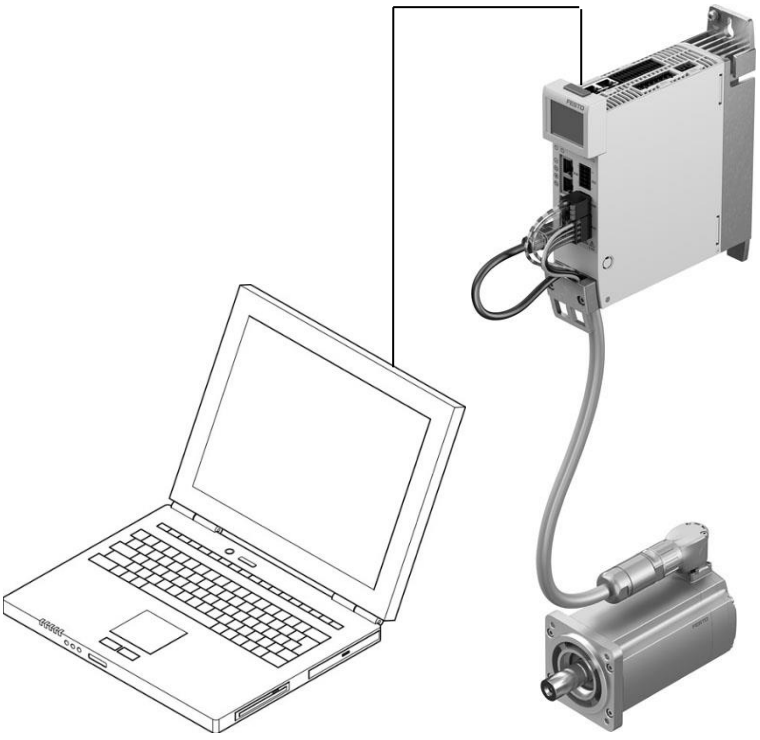


Figure 1-1 Tested system

1.2 Field configuration

Ensure that the fieldbus is configured on EtherCAT

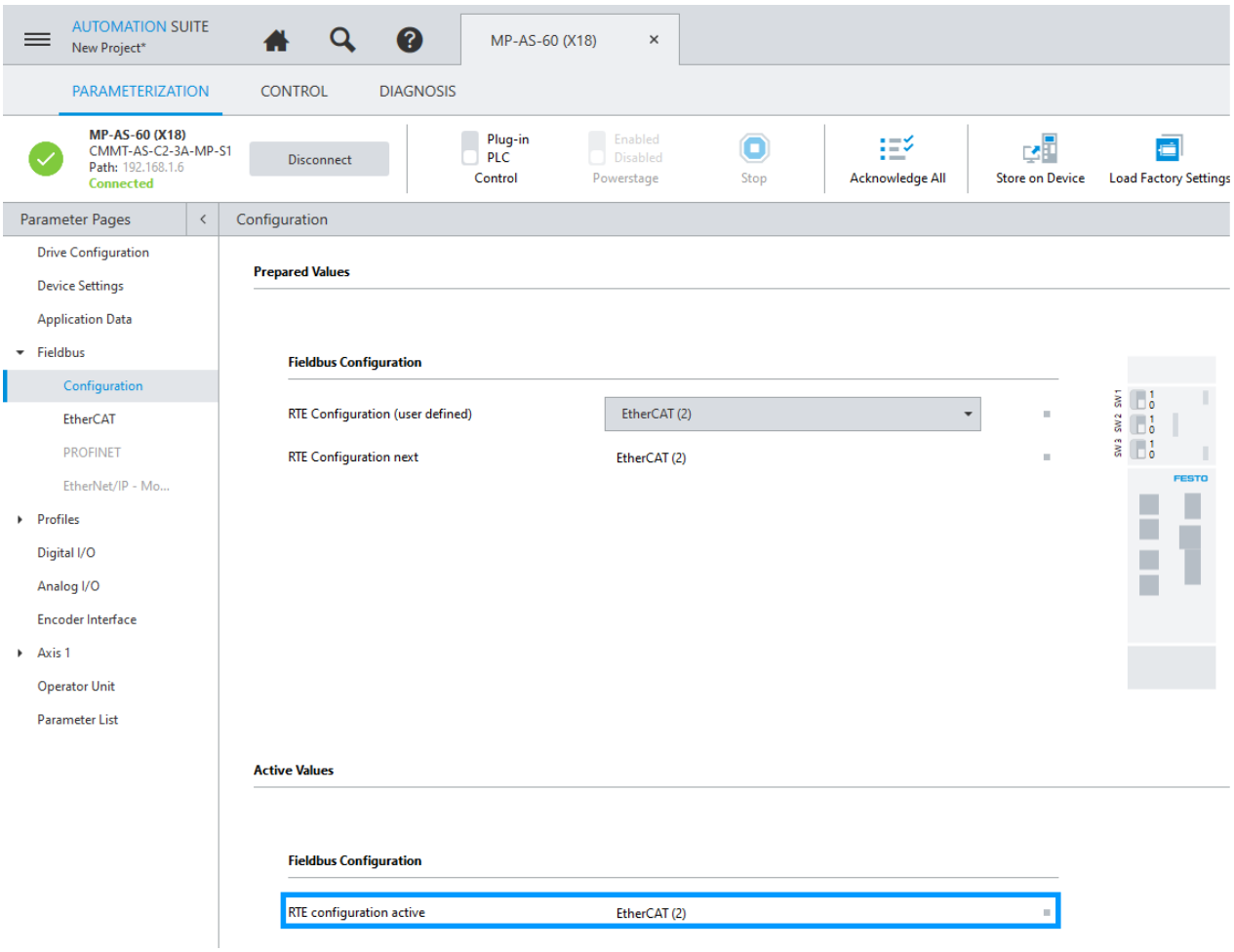


Figure 1-2 Fieldbus configuration

If it says something else, select 'EtherCAT' in the RTE Configuration (user-defined) and store the changes, then restart the device.

2 Useful links

Subject	URL
GitHub repository	https://github.com/Jkachoura/CMMT-EtherCAT
Visual Studio 2019	https://visualstudio.microsoft.com/vs/older-downloads/
Pcap	https://npcap.com/#download
SOEM	https://github.com/OpenEtherCATsociety/SOEM

3 Introduction

The examples presented in this document are based on the hardware and software mentioned in the previous chapter. This does not mean that they will not work on other versions, or other hardware. It only means they were successfully tested on the ones mentioned before. To make this Application Note more platform independent we used CMake a building platform supported on both Windows and Linux.

3.1 Preparation of the software (Windows)

Install CMake and C++ from the Tools and Features menu in Visual Studio.

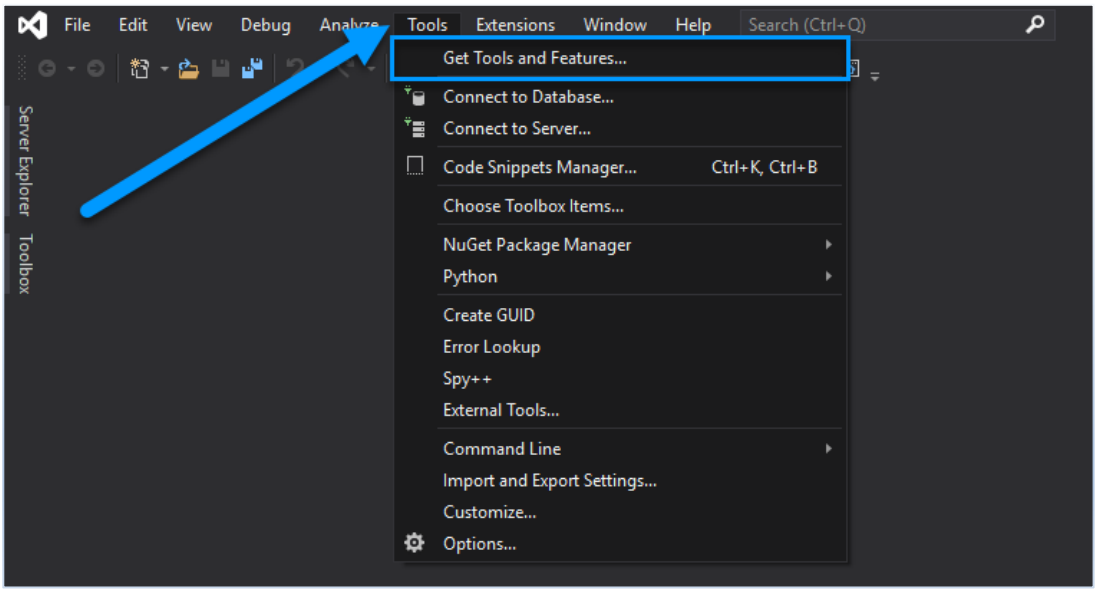


Figure 3-1 Where to find the Tools and Features menu

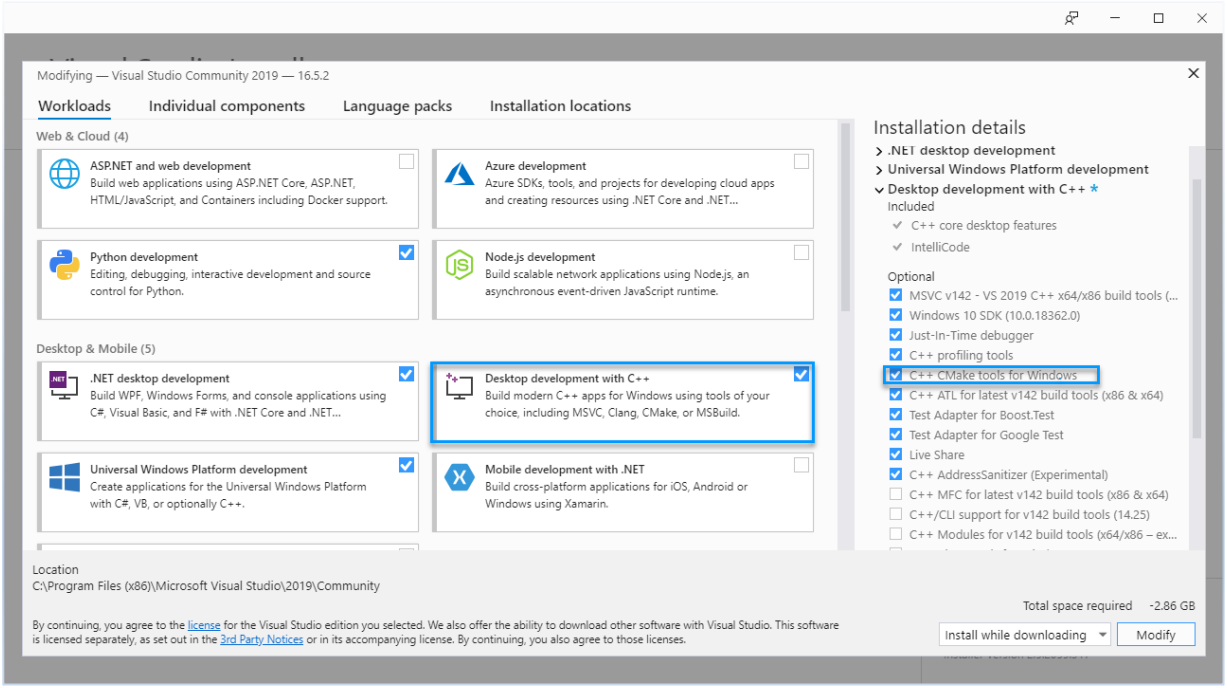


Figure 3-2 Tools and Features menu.

3.2 Install Npcap

EtherCat is not a default part of Windows or Linux. To get RAW access to the Ethernet Adapter we use the PCAP driver interface. This interface is also included in programs like WireShark, to monitor network traffic.

Download and install the program from : https://npcap.com

On most systems a restart is required after installation. All configuration of this software is done automatically.

4 Sample program

The sample program is included in this Application note. After the instructions in chapter 2, it is now possible to open a CMake folder.

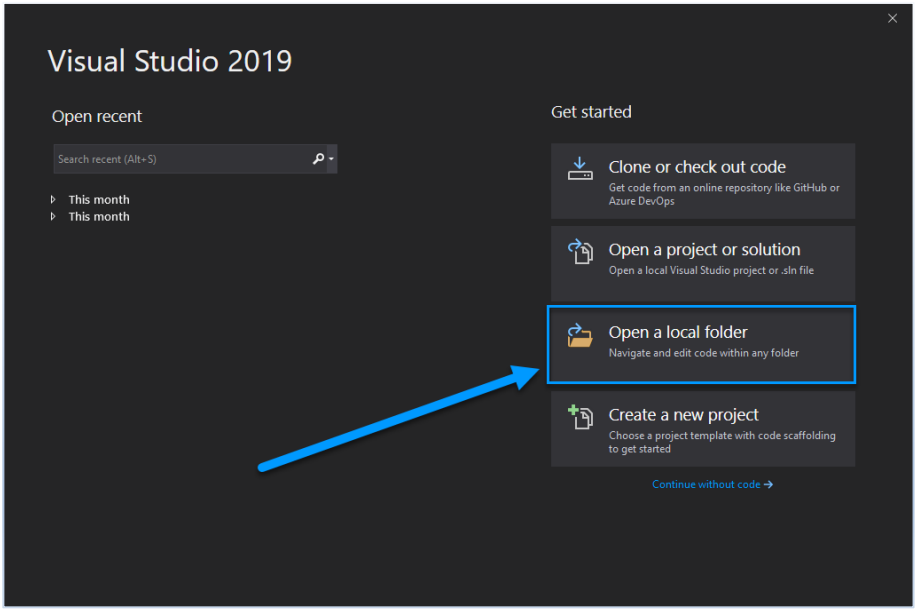


Figure 4-1 Open the sample program

4.1 Open the target view

The sample program includes multiple sample programs. The general programs to get more information about your topology and a program made by Festo to show you how to control your drive. On the right side of your screen we can change the default view, to the target view. This will display all individual programs.

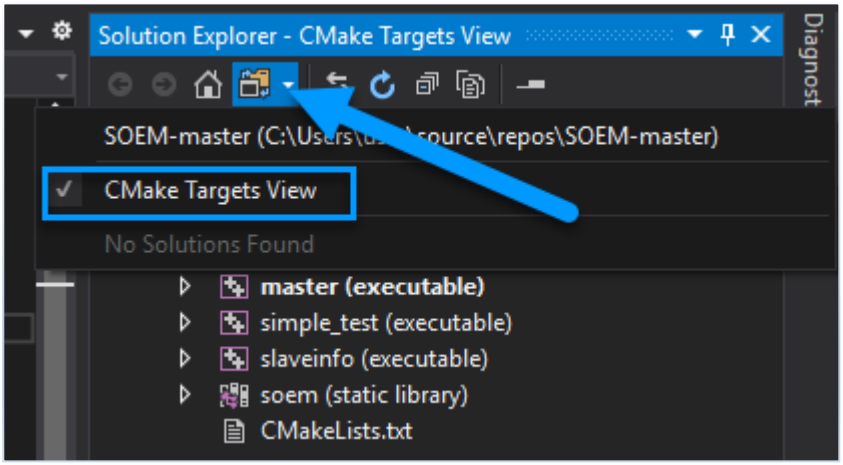


Figure 4-2 Target View

4.2 Select the slaveinfo program

The “slaveinfo” program is a very useful standard program in de SOEM library to get information about your EtherCat master and slaves. To run this program, right click on the slaveinfo (executable) and

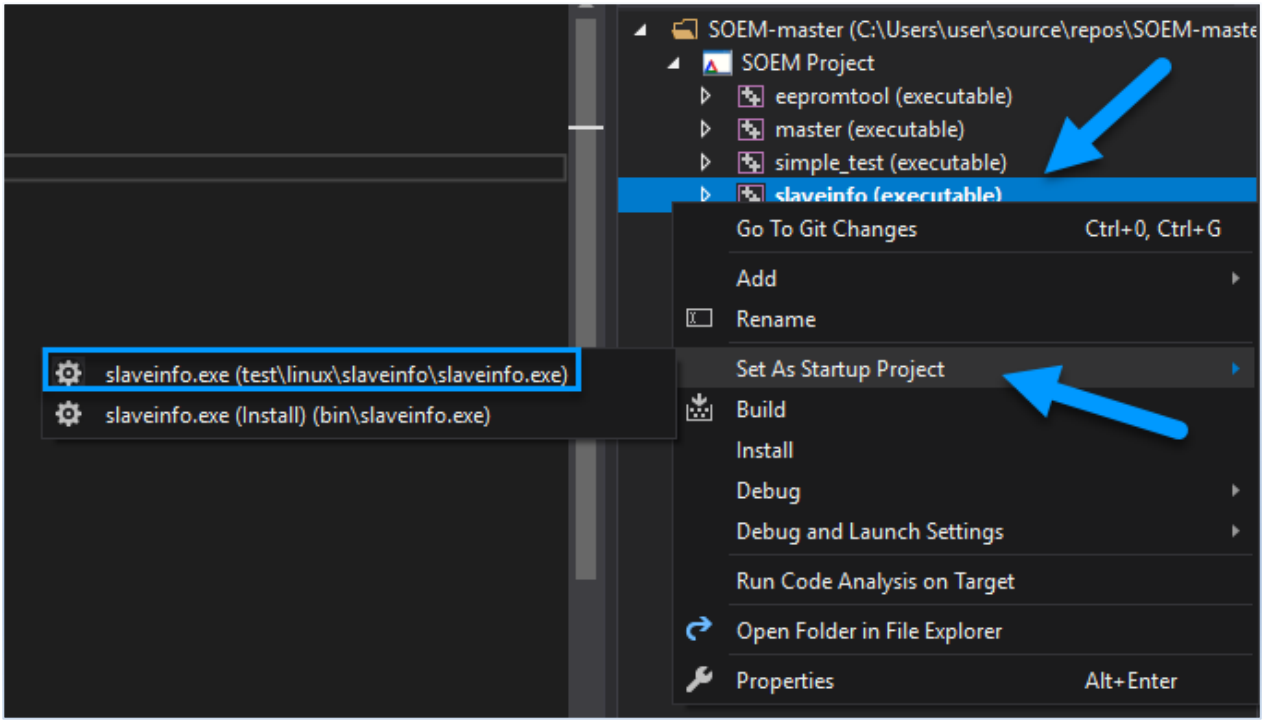


Figure 4-3 Select Startup project

4.3 Learn your network device name

The program needs to know on where to setup the EtherCat master. For this we need to have the unique identifier of your network adapter. On Linux this can be “eth0” for example. On Windows this is a longer unique description number. We can find this number with the “slaveinfo” sample program.

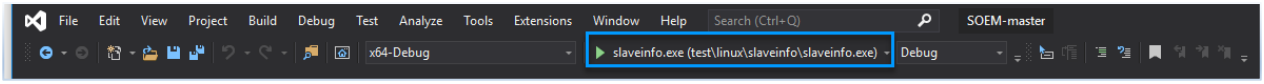


Figure 4-4 Start the program form the top bar

The output of this program will show us the unique identifying number of each connected network interface.

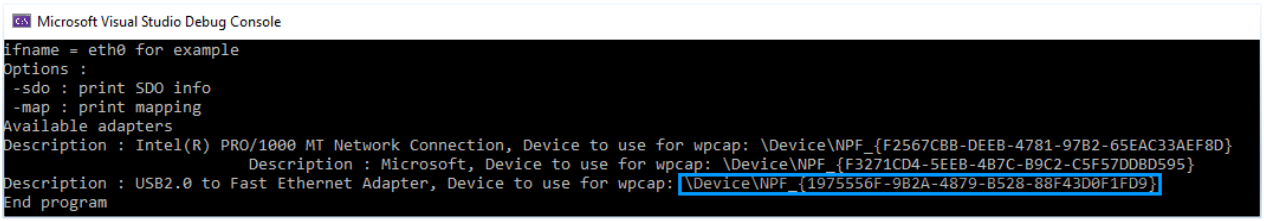
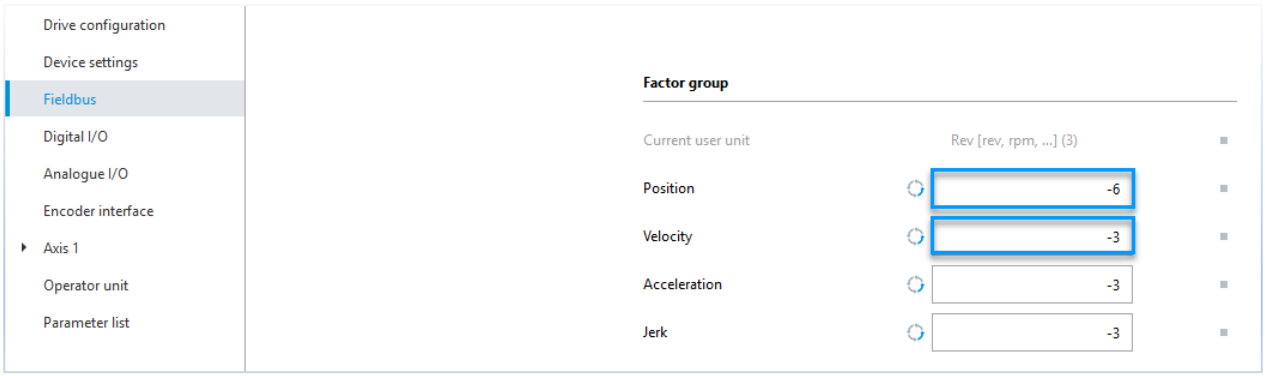


Figure 5-5 List of interfaces

In this example the interface number is "\Device\NPF_{1975556F-9B2A-4879-B528-88F43D0F1FD9}". The slash in the programming language C is however a special character. We need to modify this value to make it correctly readable by our programming language. To do this we simply add an extra “\” to every “\”. The new value now becomes \\Device\\NPF_{1975556F-9B2A-4879-B528-88F43D0F1FD9}.

5 Units of movement

The formatting of data is dependent on how it is configured in Festo Automation Suite. In this example an unlimited rotative drive is used. We configured a unit of revolution/rotation, with a precision of -6.



This means that when we want to move to a position of 20 rotations, this is transmitted over the fieldbus with this extra precision of -6. Resulting in a setpoint of 20 000 000 to achieve this.

The same is done for the Velocity, in this case we are using a precision of -3, resulting in a setting of 10 000 if we want to achieve 10 rpm.

6 The Master program

Copy the device name and select the master program, like done in chapter 3.2.

6.1 Open the main source file

Double click to open the “main.cpp” source

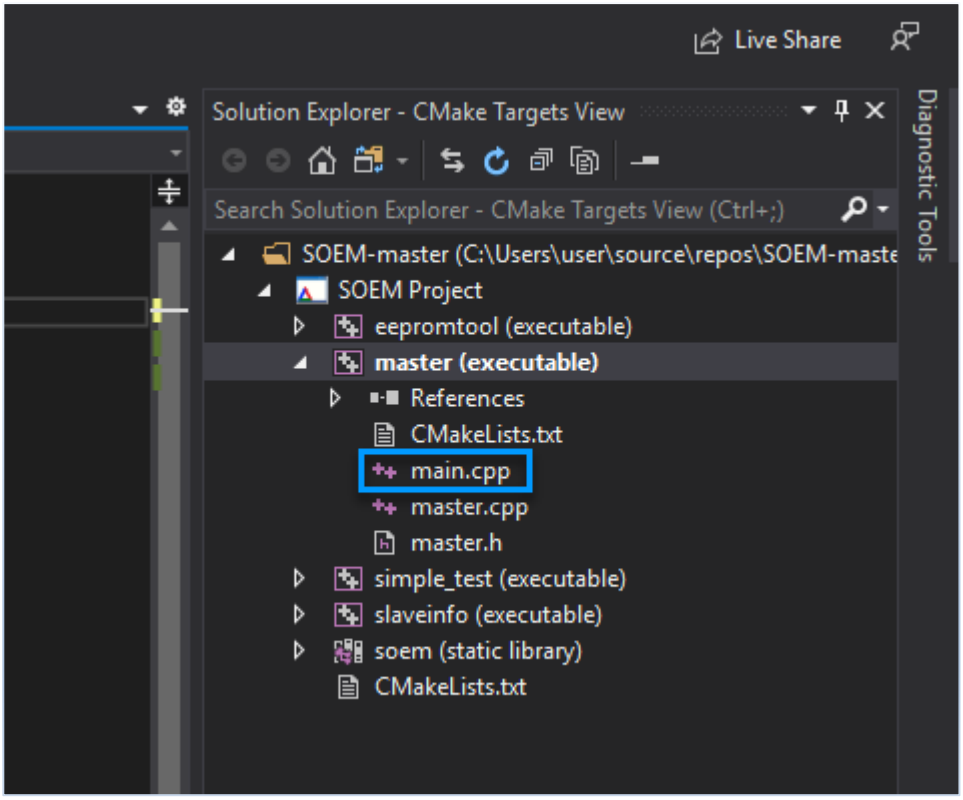


Figure 6-1 The sample program

6.2 Paste the correct network adapter

Use your unique network identifier from chapter 3.3

```
#include "master.h"
#include "slave.h"

int main(int argc, char* argv[]) {
    // Your network interface name here
    char ifaceName[] = "\\Device\\NPF_{A00D620E-09CB-4317-9A0F-4DAAD76B366E}";
    Master ecMaster(ifaceName, 8000);

    if (ecMaster.connected()) {
        Slave ecSlave(ecMaster, 1);

        ecSlave.acknowledge_faults();
        ecSlave.enable_powerstage();
        ecSlave.referencing_task();

        // Relative
        ecSlave.position_task(100000, 600000);
        ecSlave.position_task(-50000, 50000);
        // Absolute
        ecSlave.position_task(300000, 600000, true);

        return EXIT_SUCCESS;
```

Figure 66-2 Selecting the correct network card

6.3 Run the test

If you work from a virtual machine, your network adapter latency is higher than normal. A cycle time of 8 ms or 8000 nanoseconds should be safe on most recent computers.

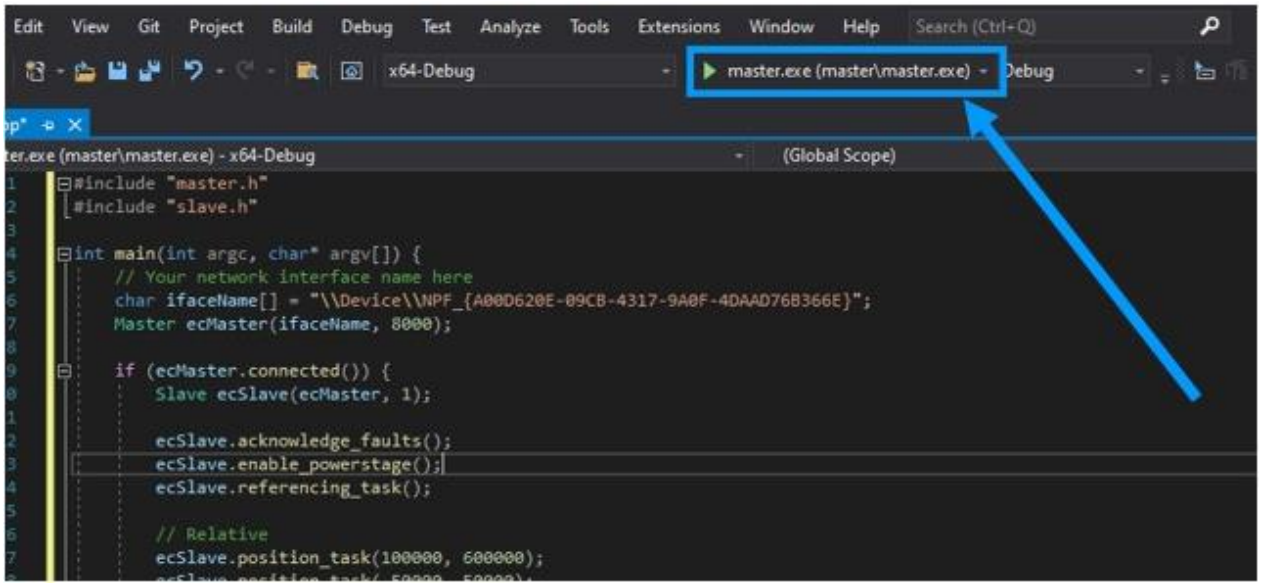


Figure 6-3 Run the program, test your configuration

If you are getting the correct amount of slaves and no errors, than the connection is successful.

7 Supported commands

For examples and usage refer to the readme of the Github page.

https://github.com/Jkachoura/CMMT-EtherCAT/blob/main/README.md#functions