QUICK START GUIDE FOR SDUECATCARD

Quick Start Guide for EtherCAT SOLAHD

There are many EtherCAT Master Systems and software applications on the market. The scope of this document is to provide the information an end user would need to add the SOLA HD SDU to their EtherCAT network. The examples in this document are specific to the Acontis EtherCAT tools, however the principles should apply to different vendor software.

Standard EtherCAT network configuration for set up:

The following image is how an EtherCAT network will appear while you are doing the initial setup. The initial set up includes scanning the EtherCAT network with a configuration tool and creating an ENI file to describe the network (more information on this in sections below).

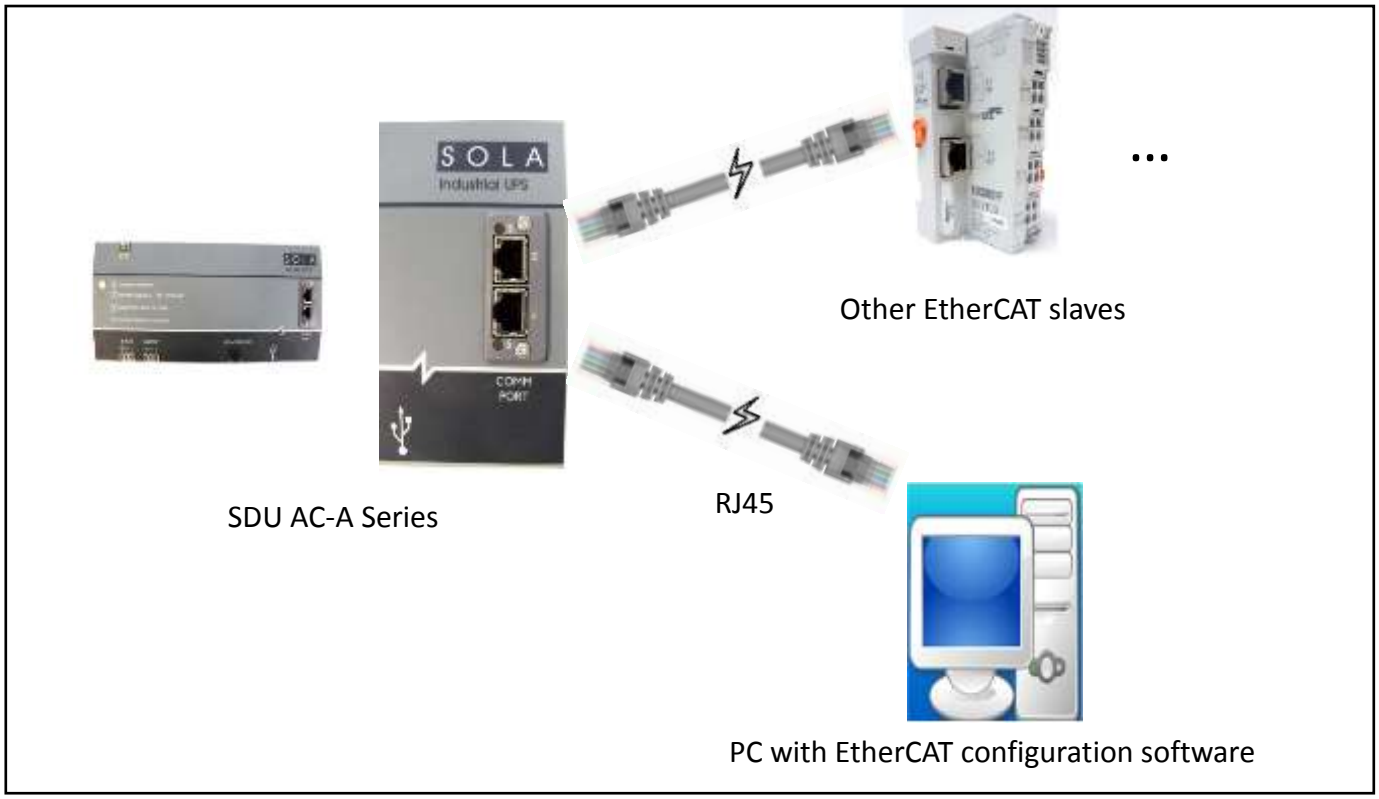

FIG.01 Diagram of Physical Setup for EtherCAT Network Configuration

Standard EtherCAT network configuration for live network:

The following image shows what a live EtherCAT network might look like once you have completed the setup. An industrial PC or EtherCAT PLC would be set up with the ENI file that describes the network.

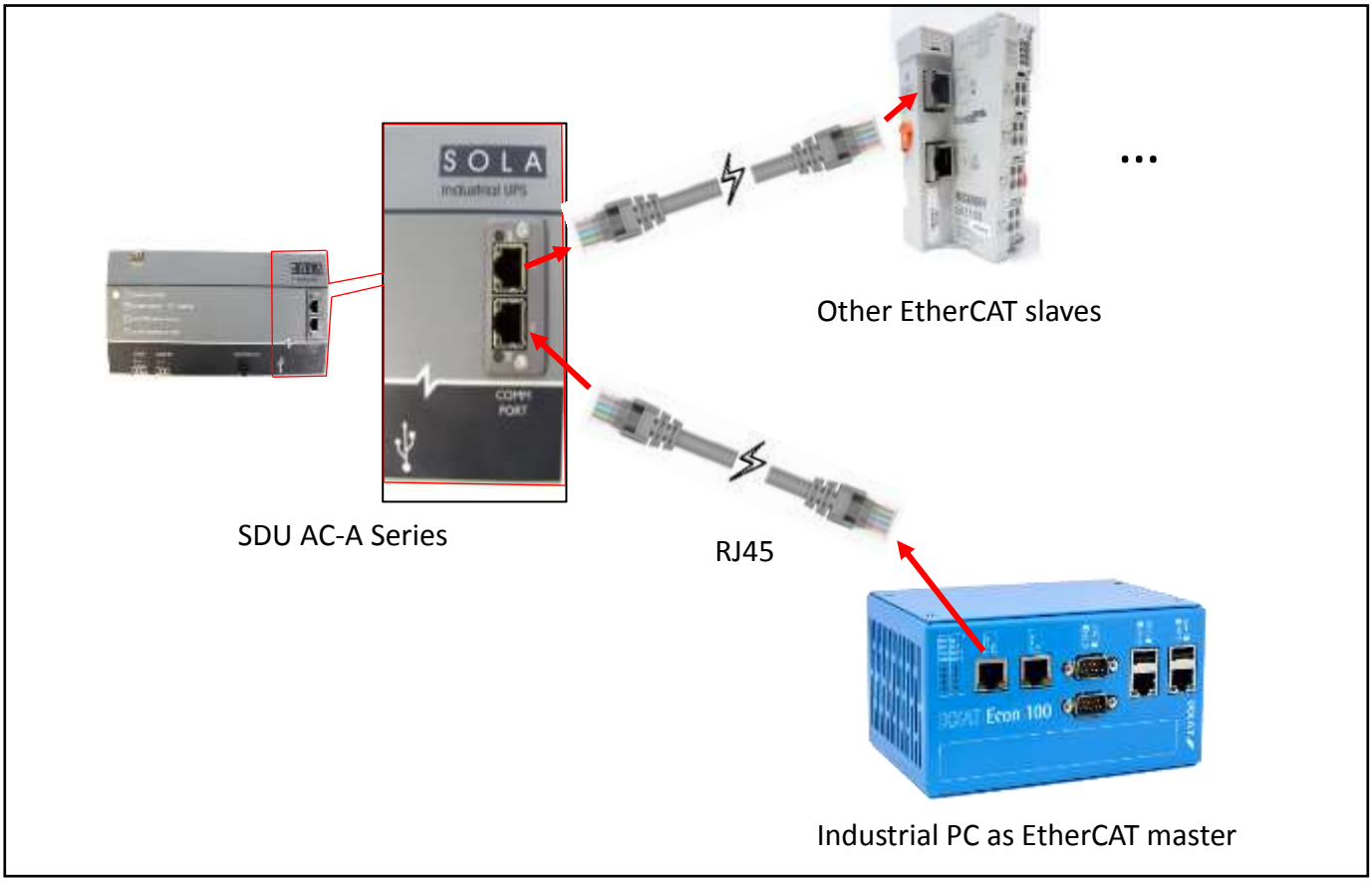

FIG.02 Diagram of Physical Setup for Live EtherCAT Network

ESI and ENI files:

There are two description files to consider when setting up an EtherCAT network. The ESI (EtherCAT slave information) is a file in XML format that describes the objects and properties of an EtherCAT slave device. The ENI (EtherCAT network file) is also in XML format, but it describes the properties of an entire network.

ESI -

This file is available for the device on the website at www.solahd.com. For most EtherCAT networks you will need to import the ESI file into the network configuration tool (example In FIG.04).

ENI -

Unlike the ESI file, the ENI file is usually generated using an EtherCAT network configuration tool. The purpose of the ENI file is to provide an EtherCAT master information about the organization of the entire EtherCAT network. Many times an EtherCAT network configuration tool can scan the network, associate the slaves with an ESI loaded into the tool and put together how everything is organized. When the tool has done this you can then export this configuration as an ENI file. The ENI file is then loaded onto the master.

Example using EC-Engineer for putting together an ENI file-

This is an example of how you would get an ENI file for your EtherCAT network using EC-Engineer in online configuration mode. As a side note, you can do offline configurations. This is not the best way to do it because you can end up with an incorrect ENI file if you are not careful.

When you open EC-Engineer, the first thing you will need to do is select “Online Configuration”. This will treat your PC as the EtherCAT master for the purposes of scanning and configuring the network.

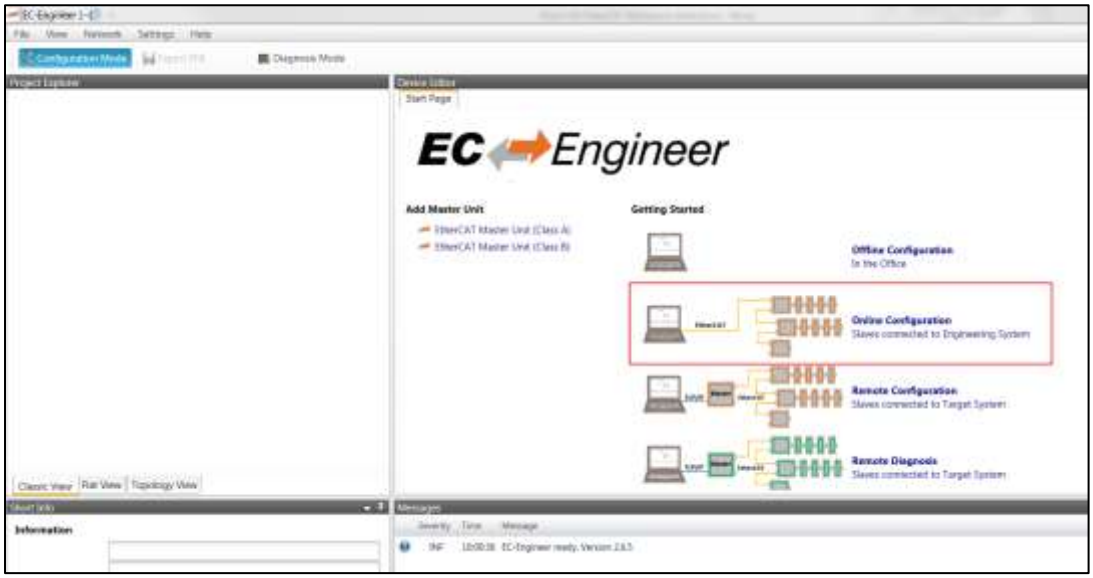

FIG.03 EC-Engineer, Online Configuration

Before you go any further you will need to bring in the ESI file of the SDU and any other slaves that will be on the network. Below is what the ESI manager looks like in EC-Engineer, this is opened from File>ESI manager. You must hit “Add file” and navigate to the ESI files you wish to import (FIG.04).

FIG.04 EC-Engineer, ESI Manager

Following this, you will select the NIC (Network interface Card) in your PC that you will use to connect to the EtherCAT network (FIG.05).

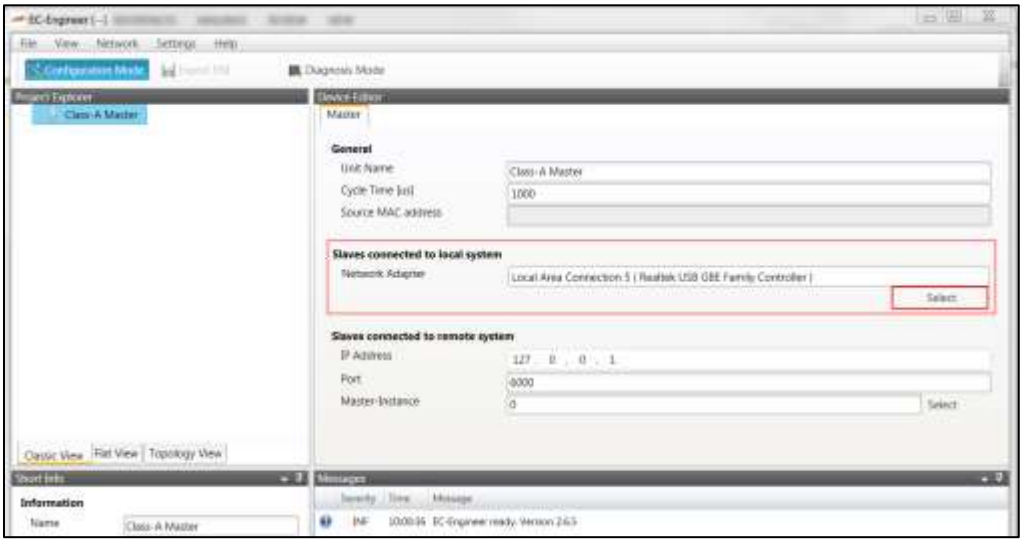

FIG.05 EC-Engineer, NIC Selection

When you select the NIC, EC-Engineer will automatically scan the network. The scan will automatically associate the ESI that you loaded with the actual device on the network.

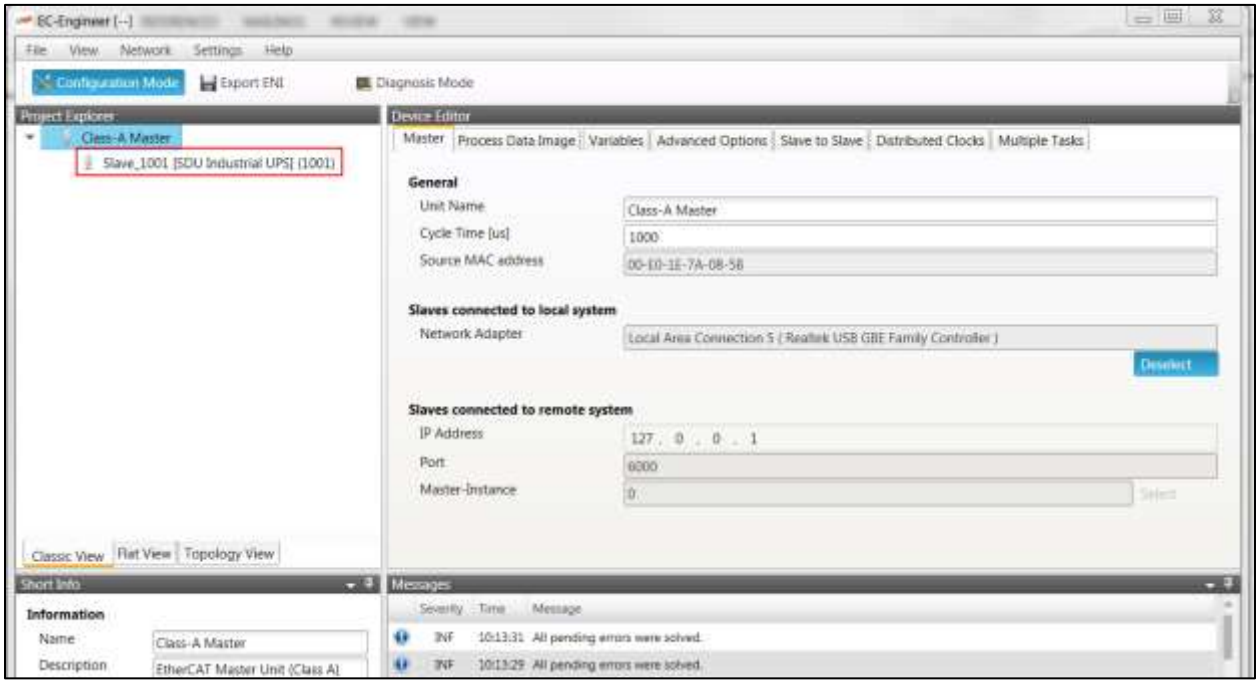

FIG.06 EC-Engineer, EtherCAT Network Scan

You can now Export the ENI by hitting the “Export ENI” button (FIG.07).

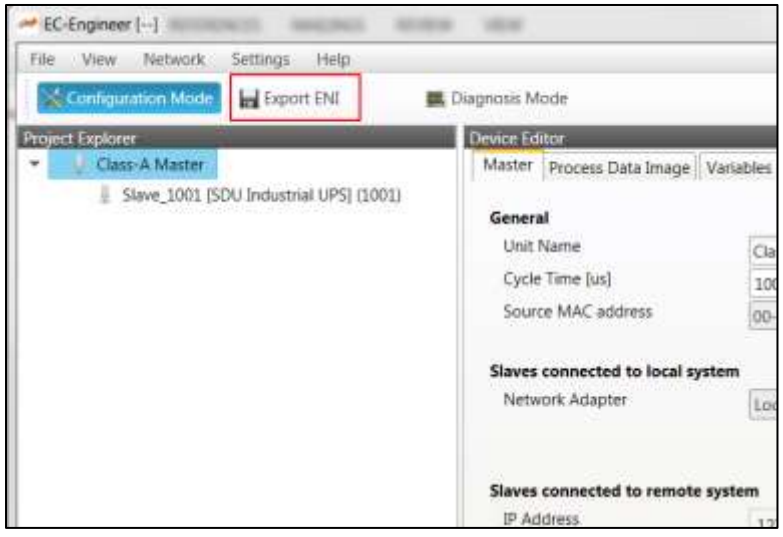

FIG.07 EC-Engineer, Export ENI File

You can also perform a quick check to view the data from the SDU (and other slaves). This is done by entering “Diagnosis Mode”. You will be asked to set the master to operational, select yes (FIG.08). The operational state in EtherCAT allows the master and slaves to start using PDOs (process data), among other EtherCAT processes.

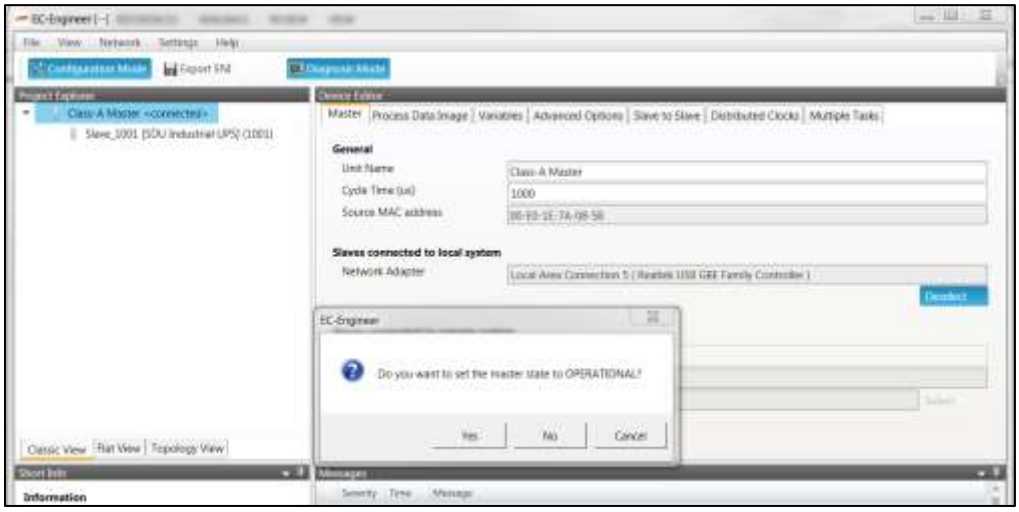

FIG.08 EC-Engineer, Setting Network to Operational

Once in Operational mode you can select the slave and view the “Variables” tab. This will show you the process data mapped variable for the SDU. By default the first 8 bytes will be Status-1, the second 8 bytes will be Status-2 and the last will be the command bytes.

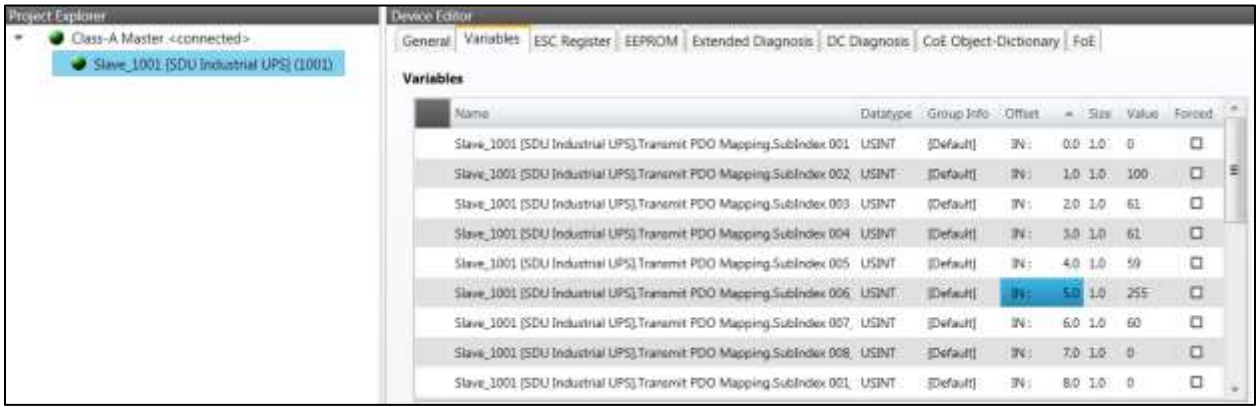

FIG.09 EC-Engineer, Data Check

Starting the EtherCAT network:

Now that you have the ENI file you can load it or reference it in the EtherCAT master. As stated before, the master will use this ENI to manage the network. The following example will show you how this works with the Acontis EC-Master evaluation software for Windows.

Please note: At this point the procedure could be quite different depending on what vendor you are using for your EtherCAT master. These vendors should have their own documentation for what exactly needs to be done with their master.

When using the Acontis Master software for windows, the user must navigate to the directory of the software shown in the windows explorer navigation bar below

(c:\…….\EC-Master-Windows\Bin\Windows\x64)

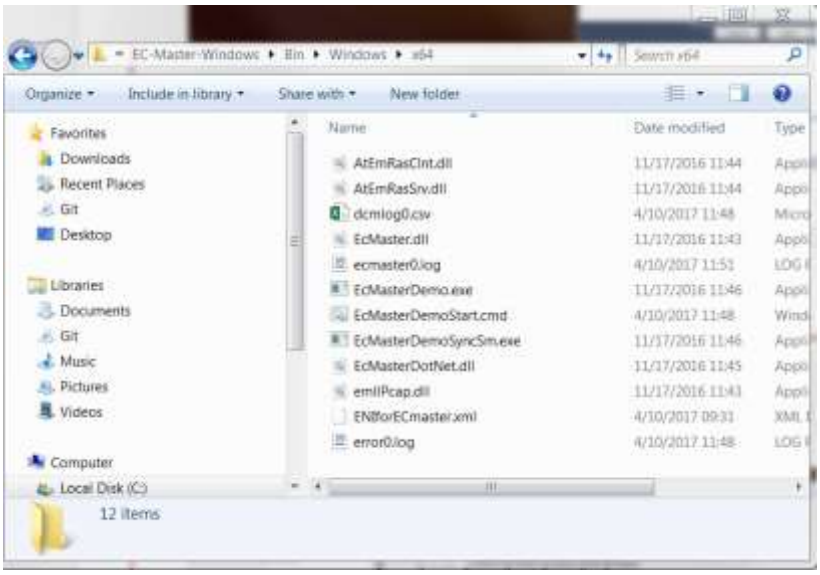

FIG.10 EC-Master Software Directory

Once in this directory, you will want to edit “EcMasterDemoStart.cmd”. In this file you must define certain parameters in order to make the EC-Master code run correctly. There are quite a few parameters, but there are only a few needed for this set up.

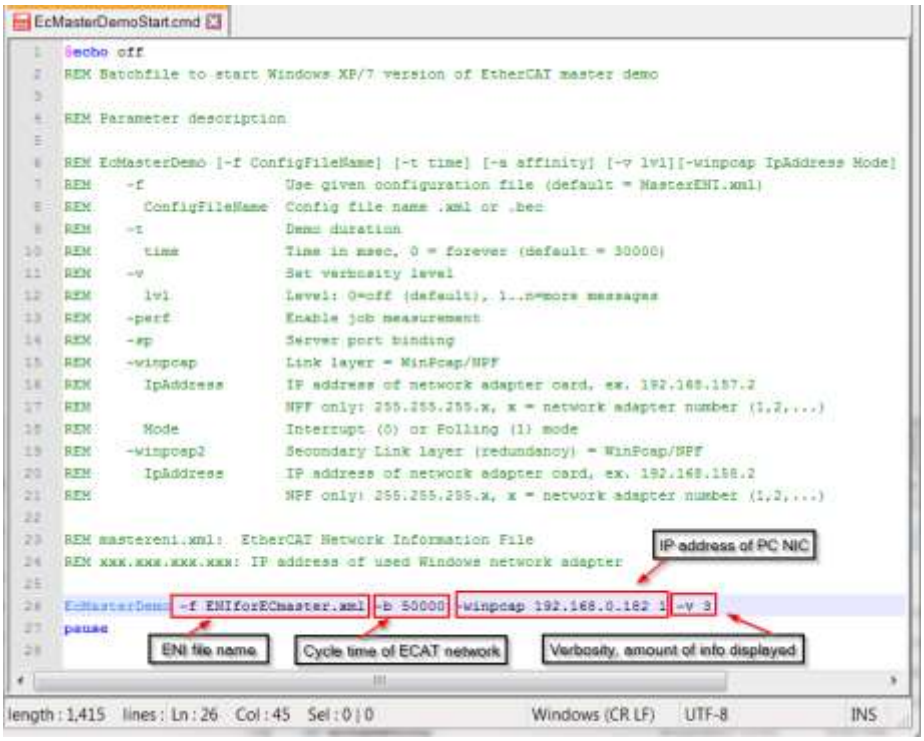

FIG.11 EC-Master, Parameters Required for Running EtherCAT Master

As can be seen in the figures above, you must copy and paste the ENI you made (in this example, ENIforECmaster.xml) and then tell the EC-Master software what it is called.

When you finish editing this you can double click the “ECMasterDemoStart.cmd” file and it will open a command prompt window that shows the software working.

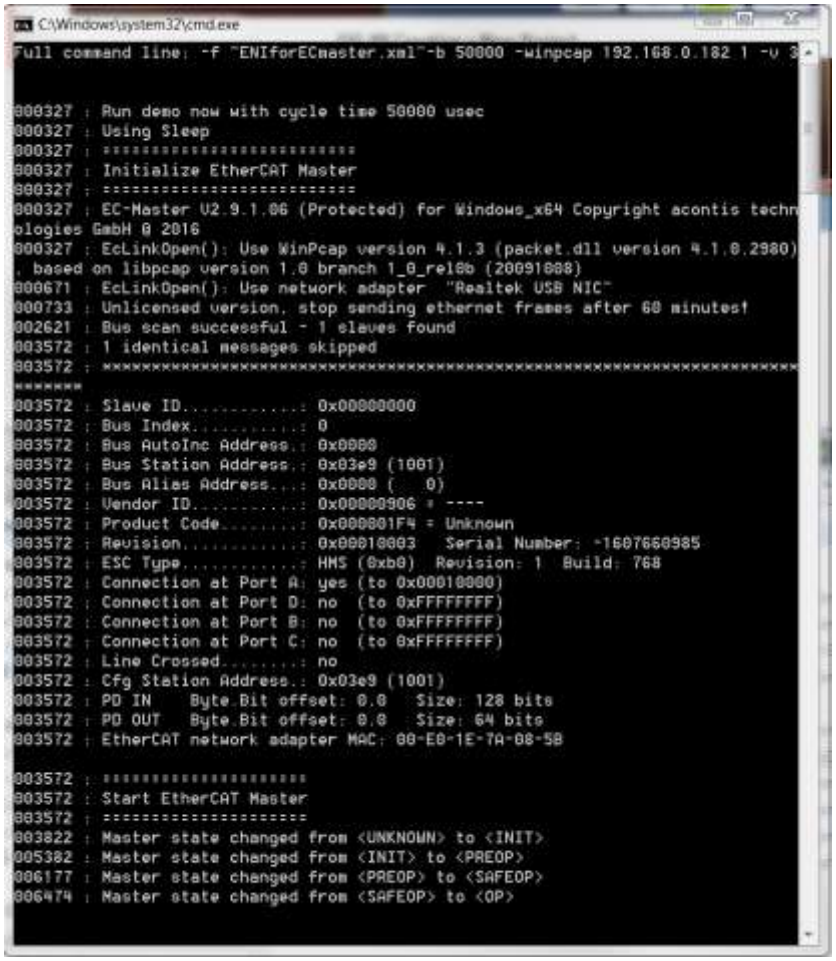

FIG.12 EC-Master, Running in Command Prompt

As can be seen in FIG.12 above, the Acontis EtherCAT Master software starts the network and eventually sets the master state/network to operational. Although no data is being displayed, the process data will actually be working. For example, EC-Engineer could be used again to do “remote diagnosis”. This would allow EC-Engineer to view the properties of the already working EC-Master software. The user could also write code to show or use the data from the EC-Master software. That is outside of the scope of this example.

END

A272-311 Rev. 0 01/2018