

# Kyland MaVIEW IDE & RTE Example

## How-to develop a Modbus TCP Server program

!!! In case of support needed, feel free to reach us out directly !!!

[maview22@kyland.com](mailto:maview22@kyland.com)

Google Chat: [maview.kyland@gmail.com](mailto:maview.kyland@gmail.com)

Prerequisites:

1. a Win7/Win10 PC (**Host\_PC**) with >2GB free space to install MaVIEW IDE (Integrated Development Environment) and to develop the virtual PLC program
2. a x86/x64 based Linux PC (**Target\_PC**) with >500MB free space to operate MaVIEW RTE (Run Time Environment) and execute the virtual PLC program developed from 1, so the **Target\_PC** would work as a PLC
3. connection between **Host\_PC** and **Target\_PC**

The whole process may take 2-3 hours in total:

**A. Install MaVIEW RTE in a x86/x64 Linux PC (**Target\_PC**) (10mins)**

**B. Install MaVIEW IDE in a Win7/Win10 (**Host\_PC**) and develop the virtual PLC program (30-60mins)**

\* a simple 'Modbus-TCP Server' program example is illustrated in this example

**C. Test the virtual PLC program (10mins)**

**D. Fill up the feedback form and attach your program (10mins)**

**Appendix. Install and setup Ubuntu Linux (60-90mins)**

## A. Install MaVIEW RTE in a x86/x64 Linux (Target\_PC) (10mins)

maview-rte.tgz download link: <https://www.kyland.com/maview22/maview-rte.tgz>

1. download `maview-rte.tgz` directly from **Target\_PC** console,

```
kydemo@ubuntu:~$ wget https://www.kyland.com/maview22/maview-rte.tgz
```

or use a USB pen drive to copy the file from **Host\_PC** to **Target\_PC**,

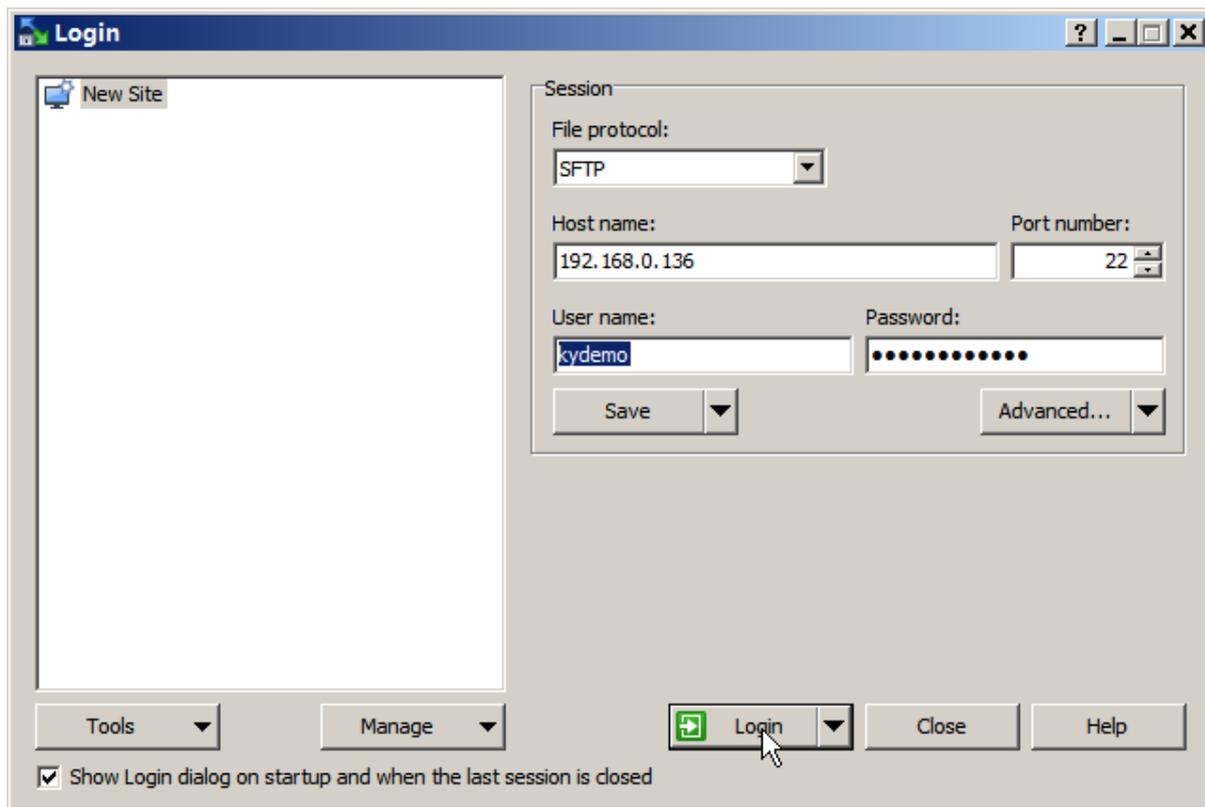
or use a SFTP software to transfer `maview-rte.tgz` from **Host\_PC** to **Target\_PC**

a. check if **Target\_PC** has 'openssh-server' package installed, otherwise install it (internet accessibility is needed)

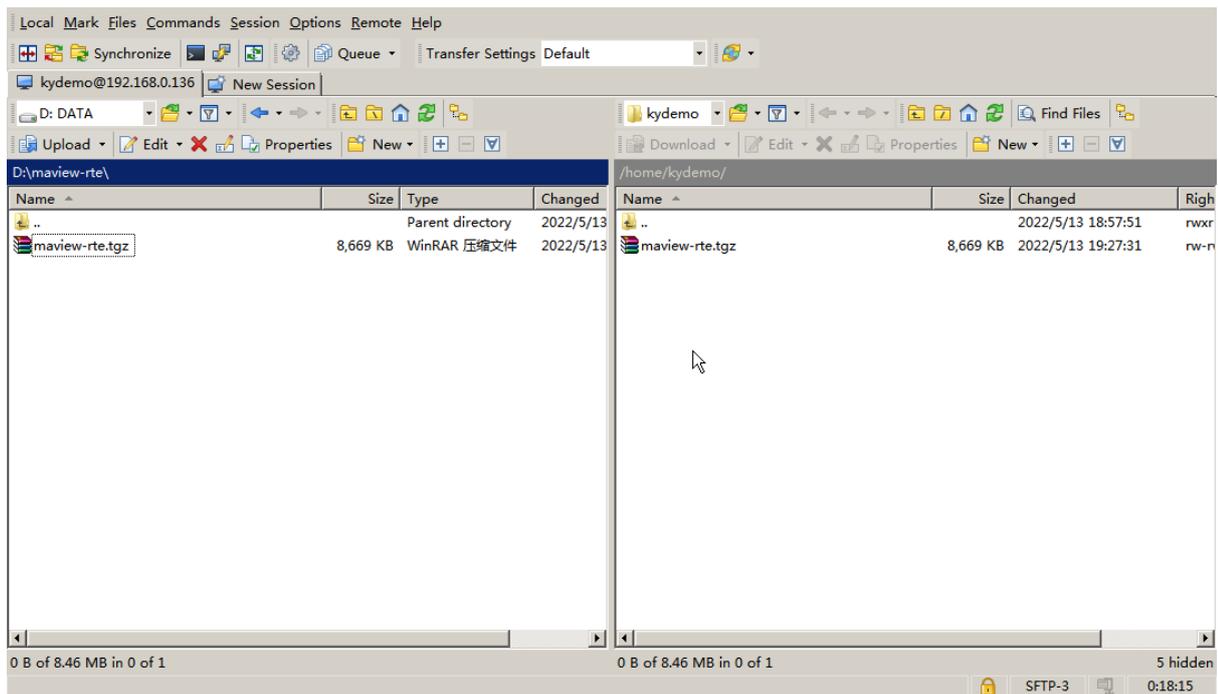
```
kydemo@ubuntu:~$ apt-get update
```

```
kydemo@ubuntu:~$ apt-get install openssh-server
```

b. use WinSCP (freeware) in **Host\_PC** to connect to the **Target\_PC** (e.g. 192.168.0.136)



c. transfer `maview-rte.tgz` to the **Target\_PC**



2. in **Target\_PC**, `untar` (unzip) `maview-rte.tgz` and execute the `installrte` script and then reboot (so to auto-execute `LinuxRTE` program at start-up)

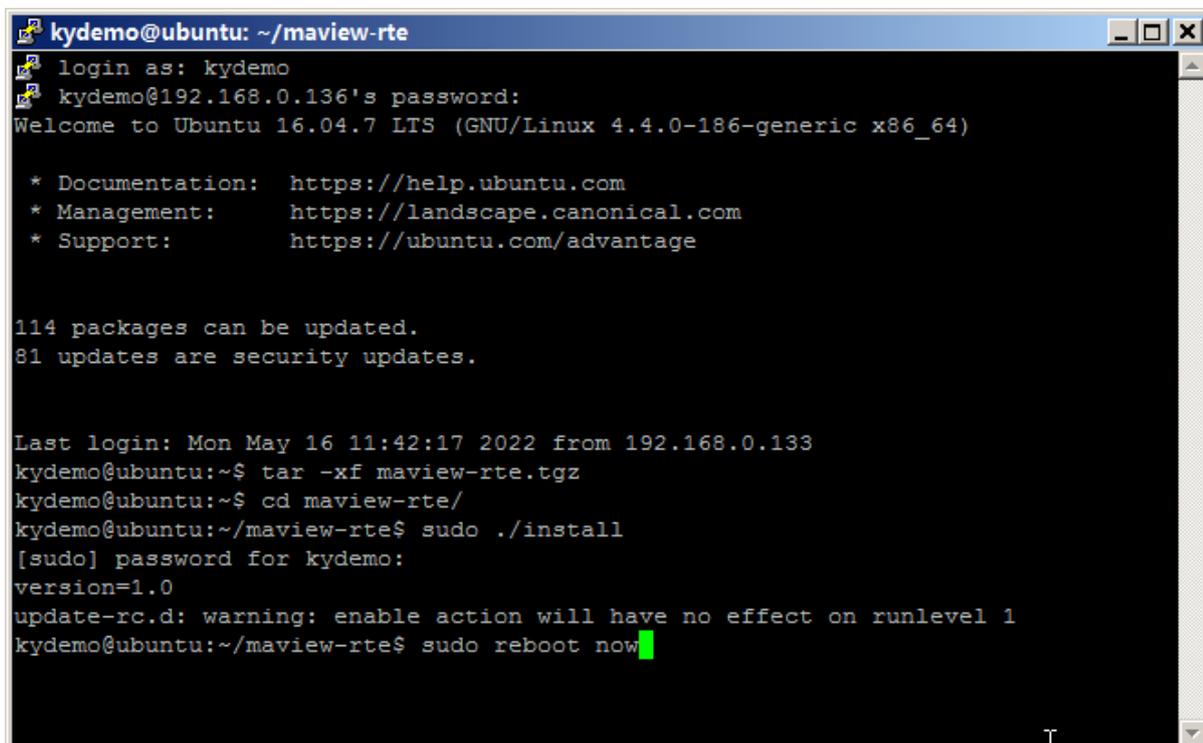
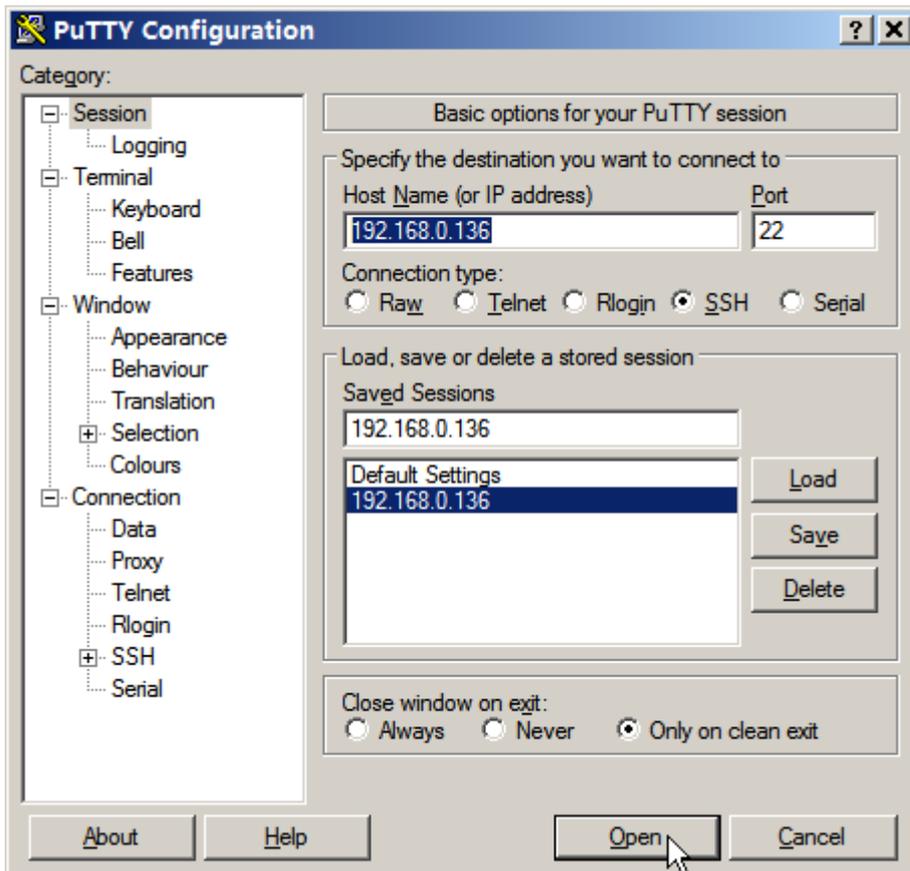
```
kydemo@ubuntu:~$ tar -xf maview-rte.tgz
```

```
kydemo@ubuntu:~$ cd maview-rte/
```

```
kydemo@ubuntu:~$ sudo ./installrte
```

```
kydemo@ubuntu:~$ sudo reboot
```

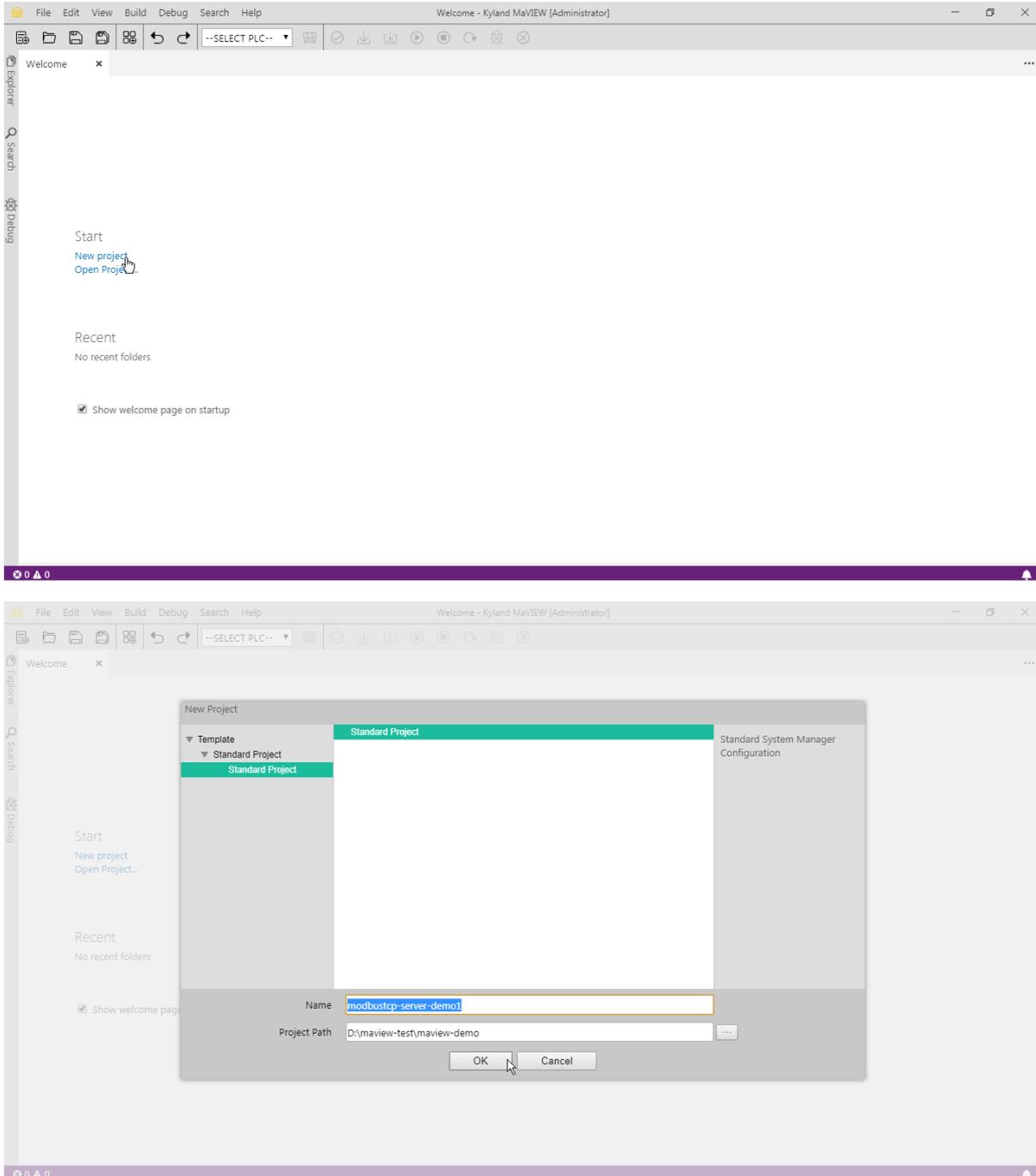
\* if you prefer to operate everything from your **Host\_PC**, you may a SSH terminal tool like 'putty' (freeware) to connect from **Host\_PC** to **Target\_PC**



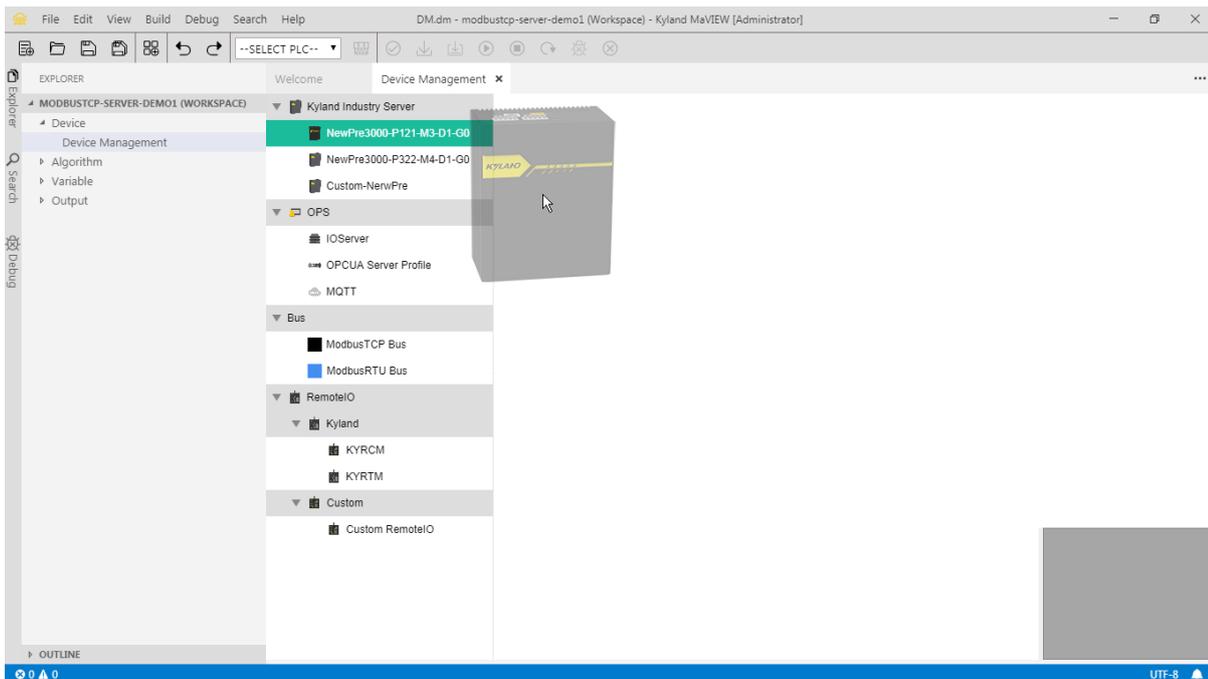
## B. Install MaVIEW IDE in a Win7/Win10 (Host PC) and develop the virtual PLC program (30-60mins)

\* a simple 'Modbus-TCP Server' program example is illustrated in this example

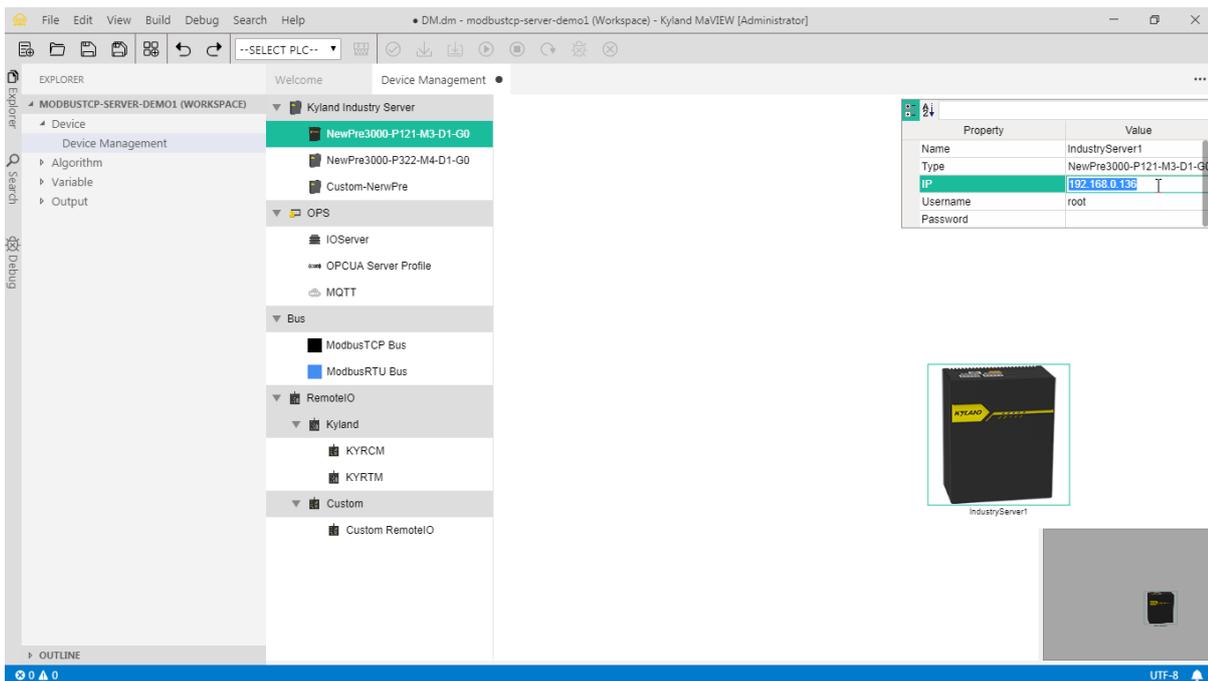
### 1. start a 'New project'



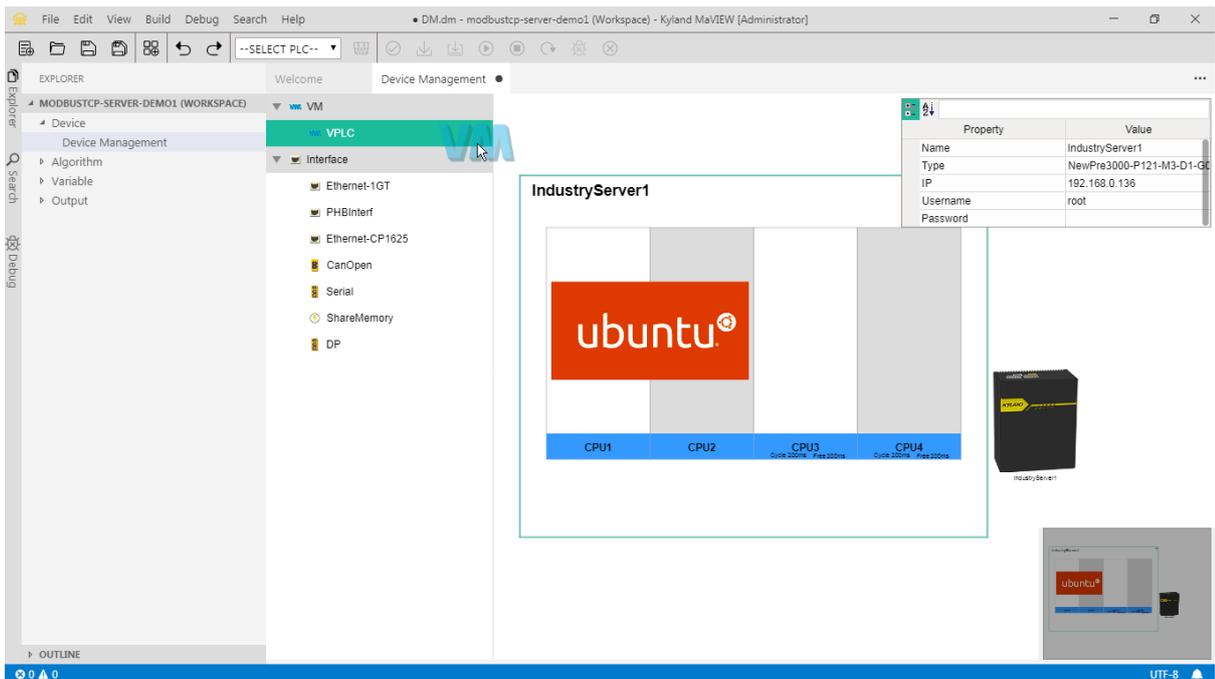
2. drag a device from 'Device Management' into the work area (if there's only standard I/O interface like Ethernet in **Target\_PC**, simply use any device from the 'Kyland Industry Server' list)



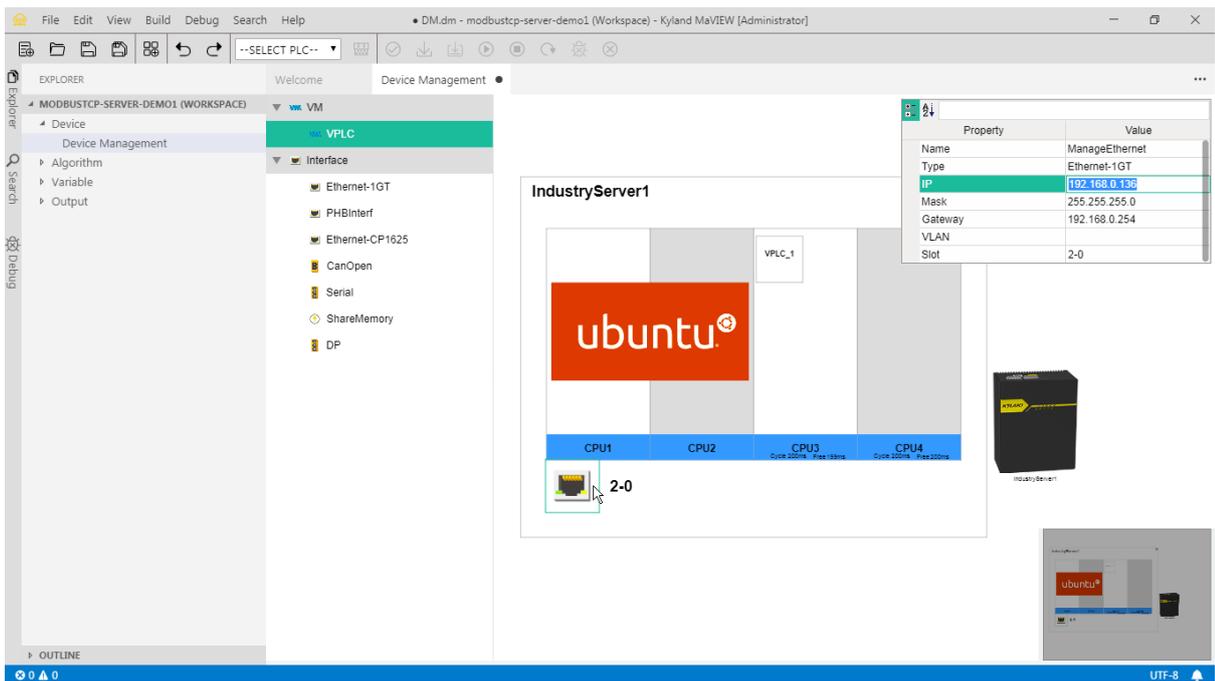
### 3. change the IP address of the **Target\_PC**



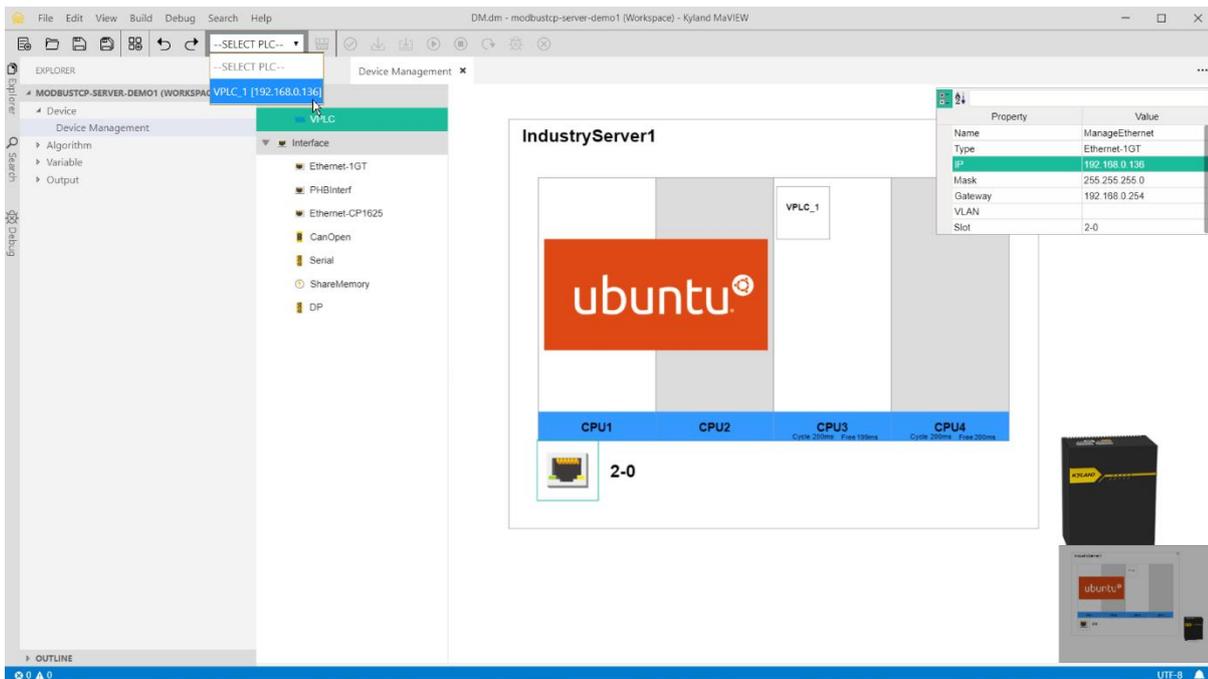
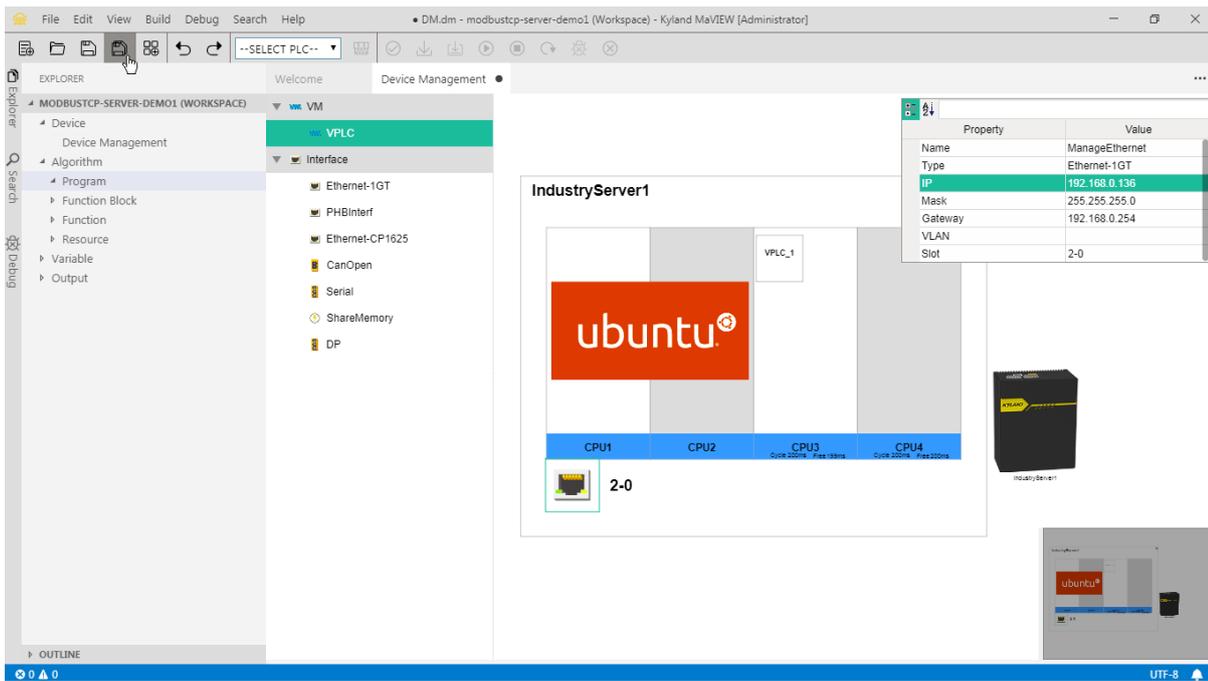
### 4. double click the new device to launch the CPU view and drag a virtual PLC into a virtual CPU



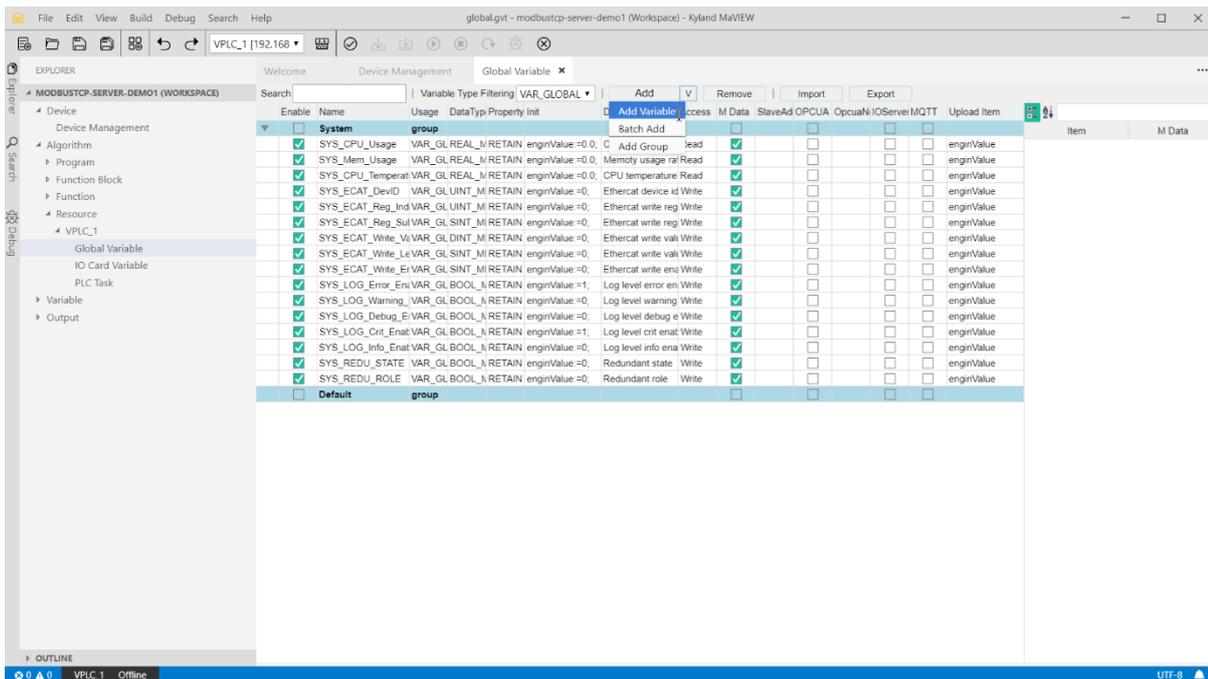
## 5. change the IP address of the virtual PLC



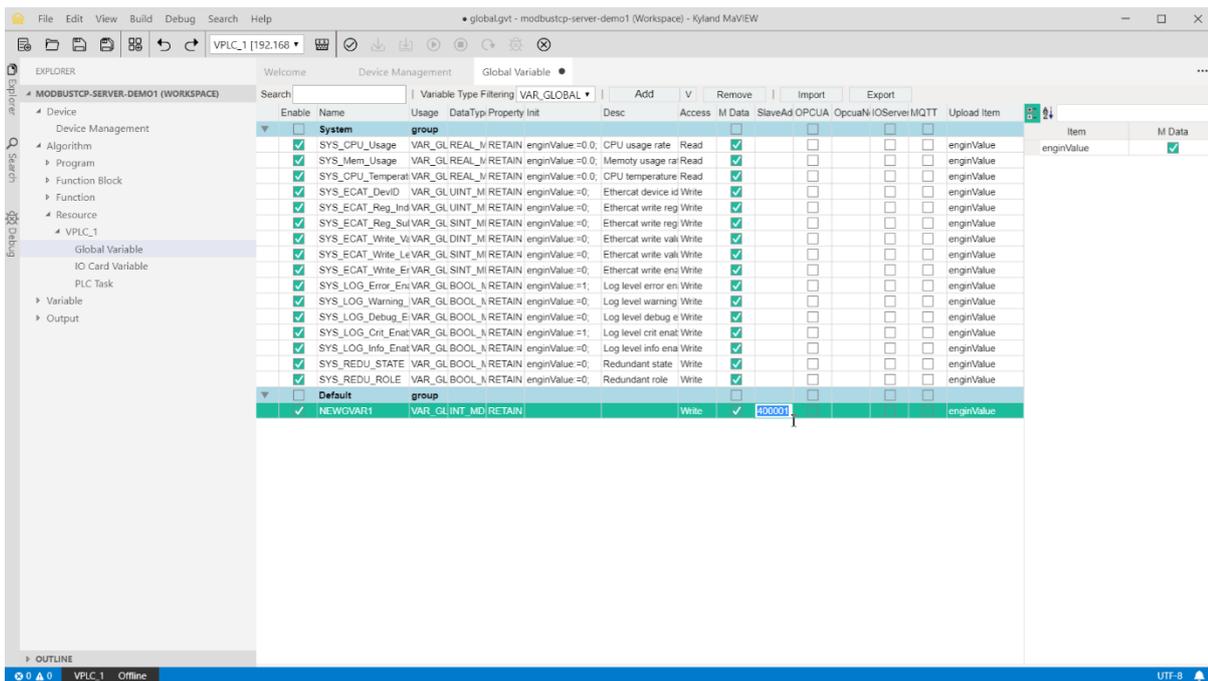
## 6. 'Save All' to save the project and select the new virtual PLC



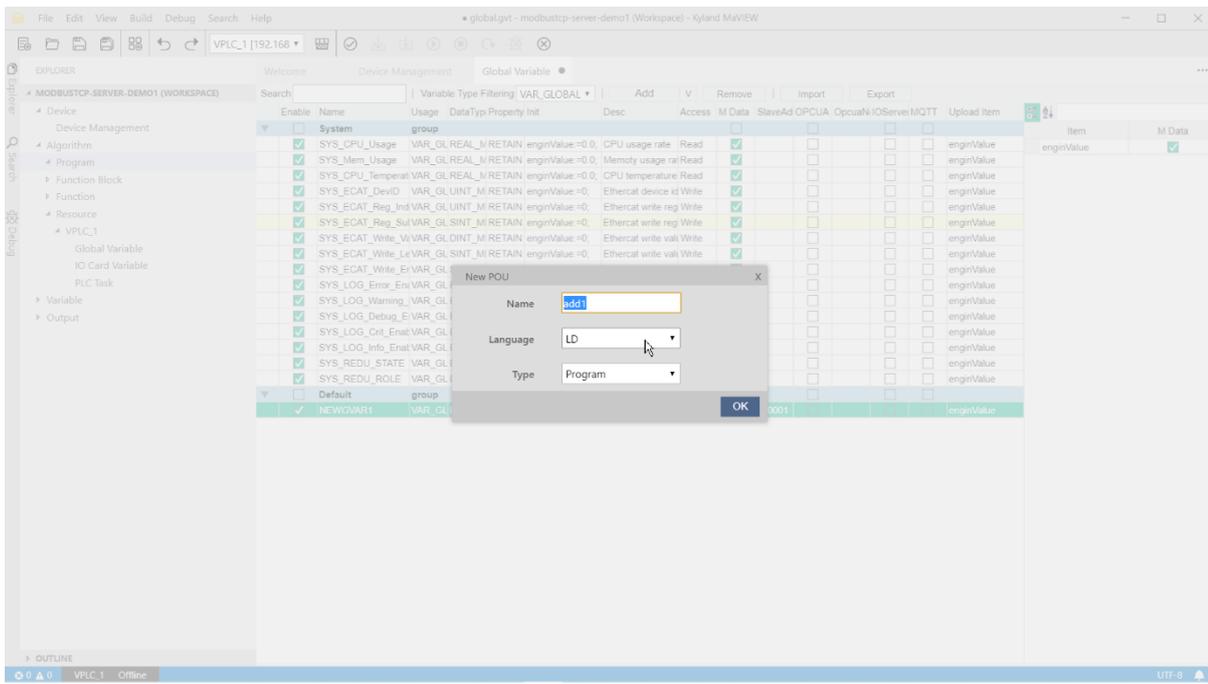
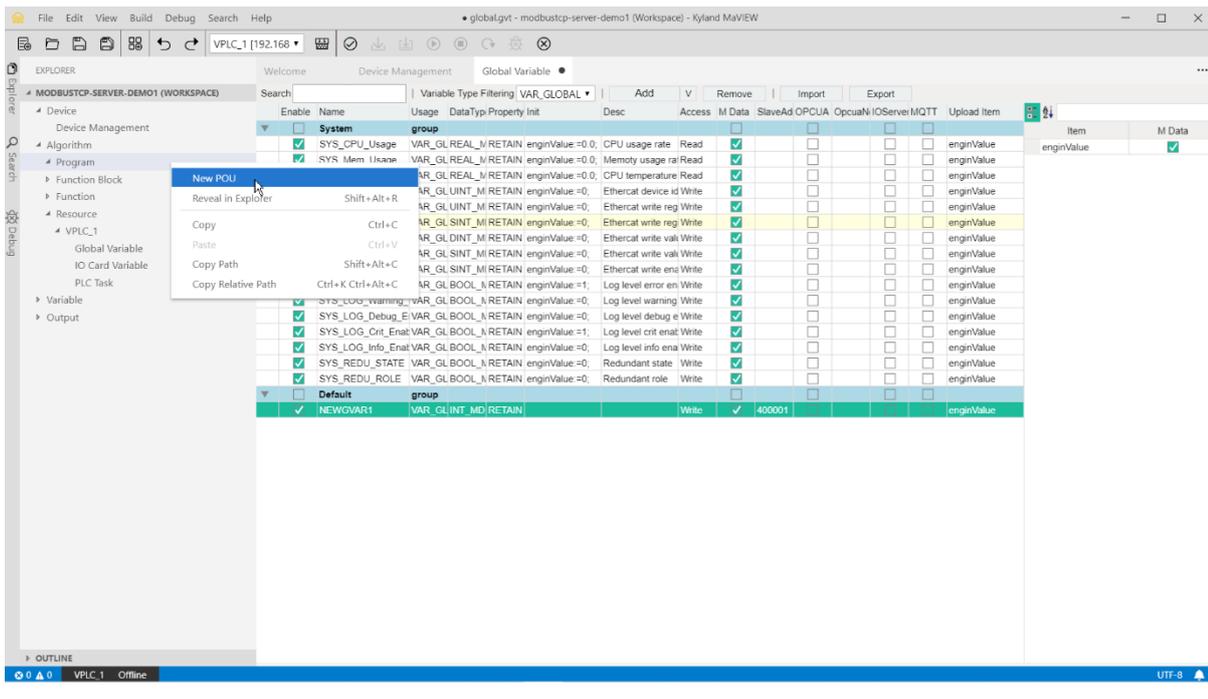
7. under the 'VPLC\_1', 'Global Variable' panel, add a new variable via 'Add Variable'



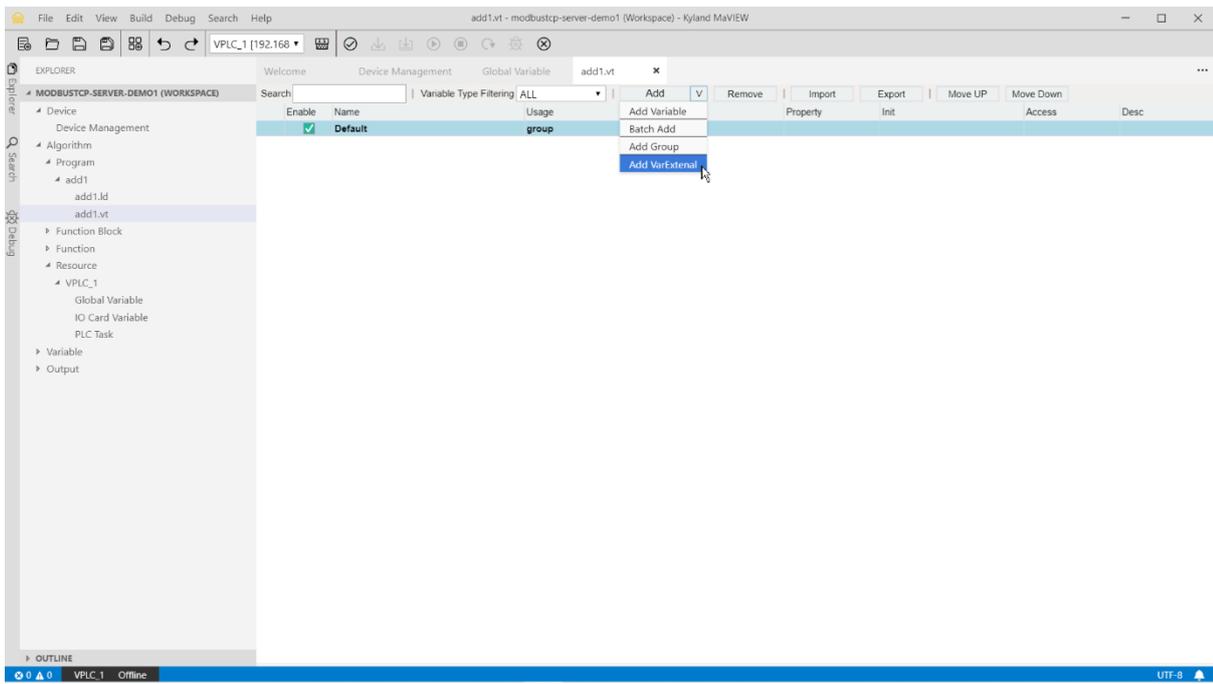
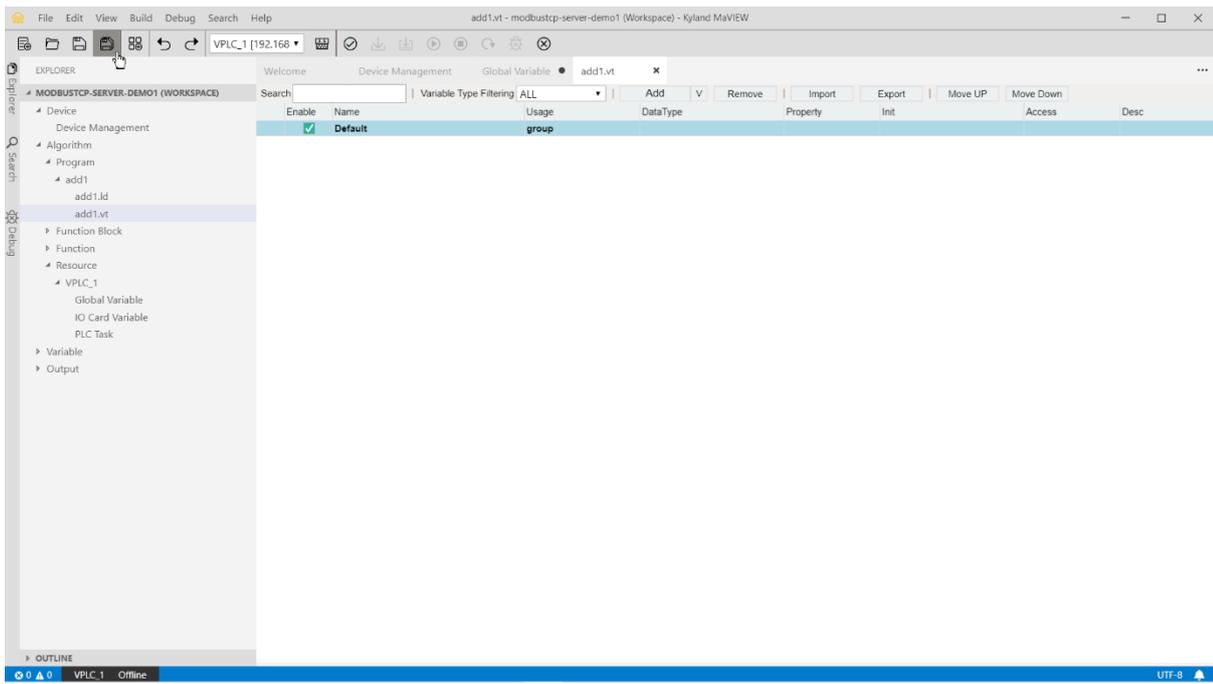
## 8. change the 'SlaveAddr' to '400001'

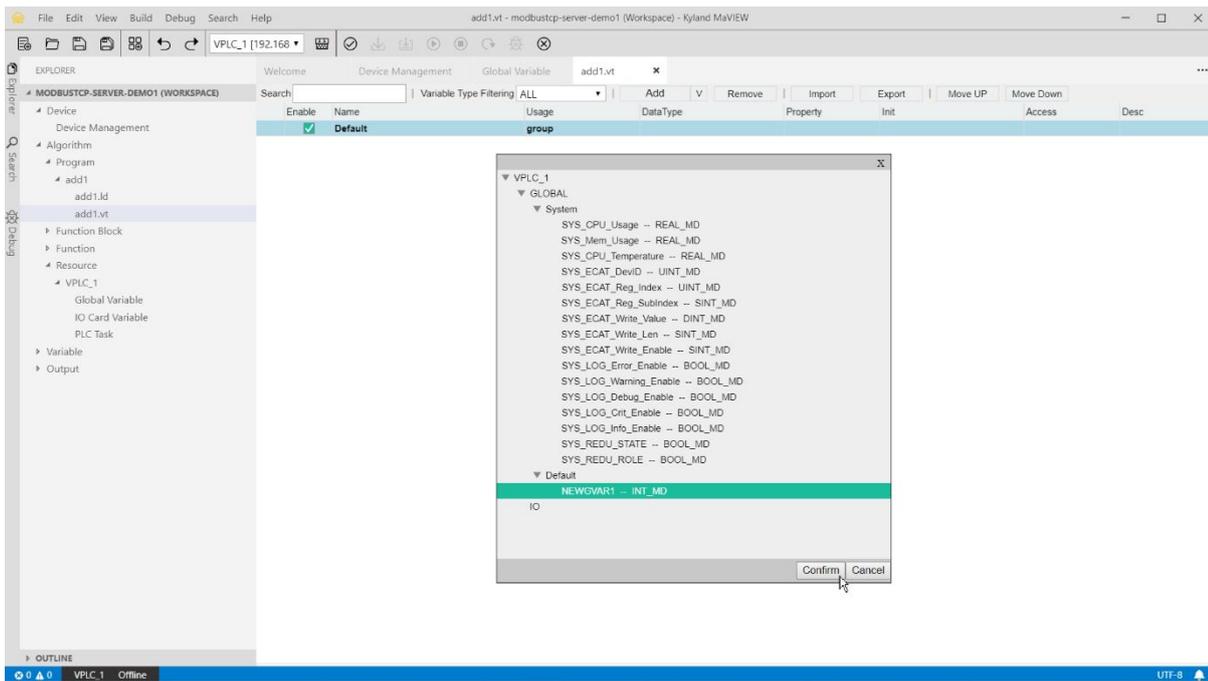


## 9. under the 'Program' panel, add a new POU with 'LD' language

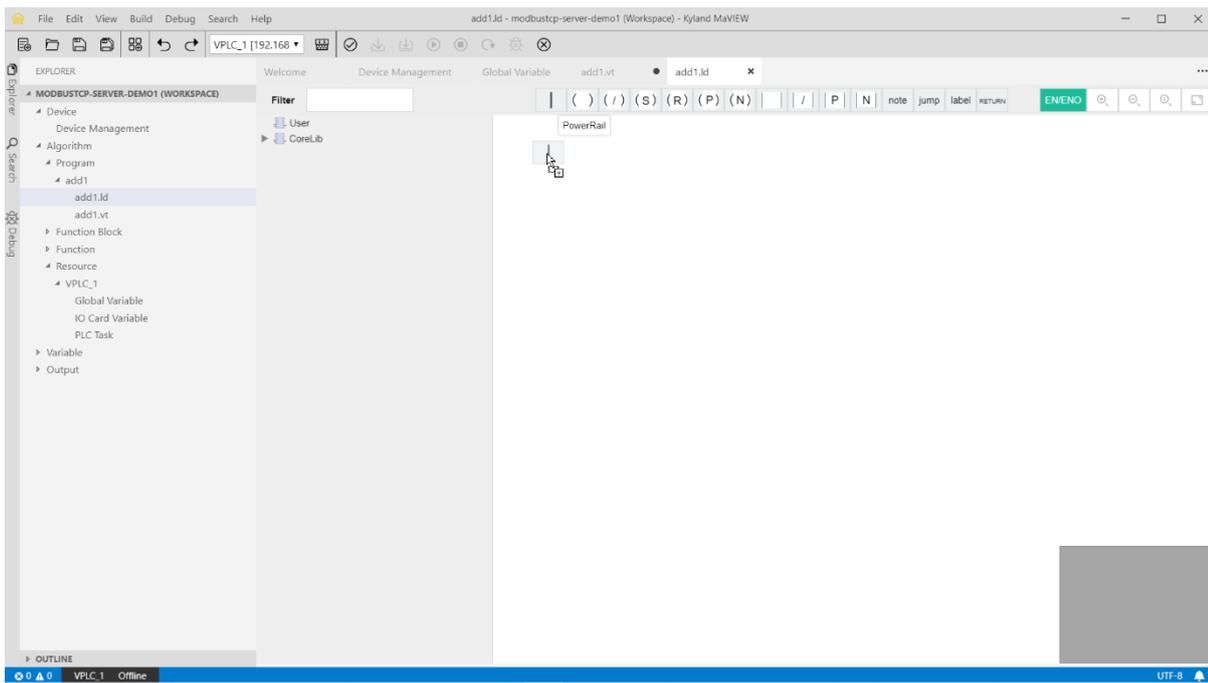


10. 'Save All' and add an external variable via 'Add VarExternal', select the attribute as the Global Variable just added before

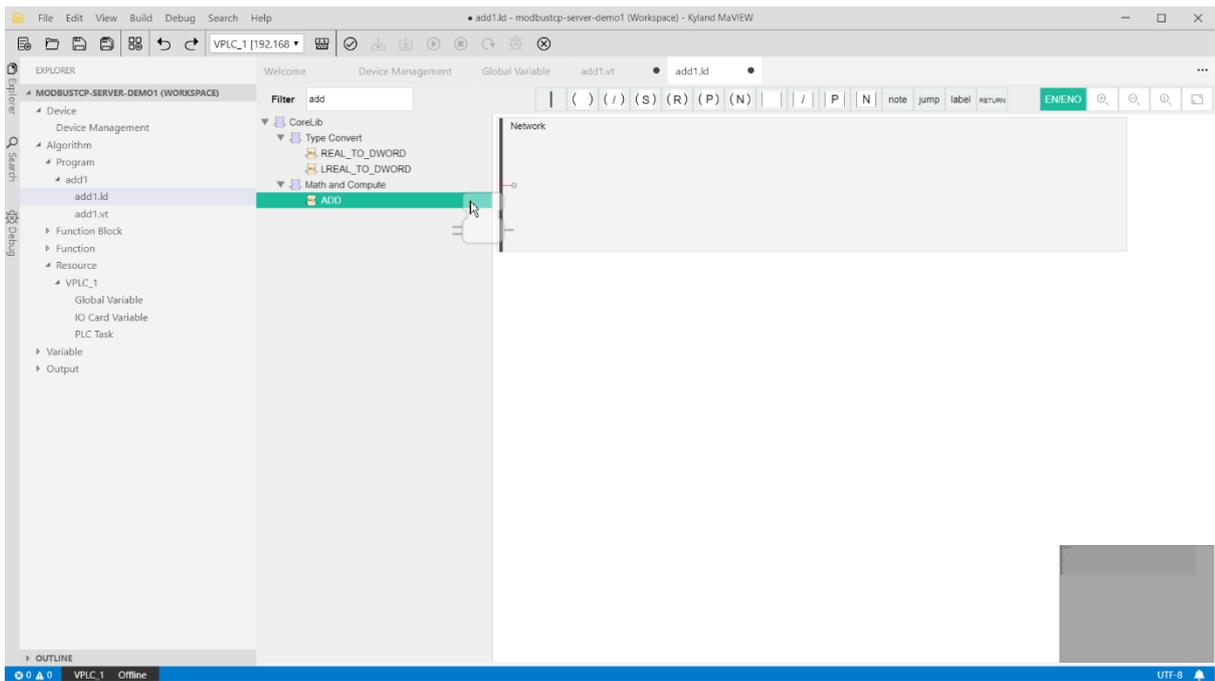




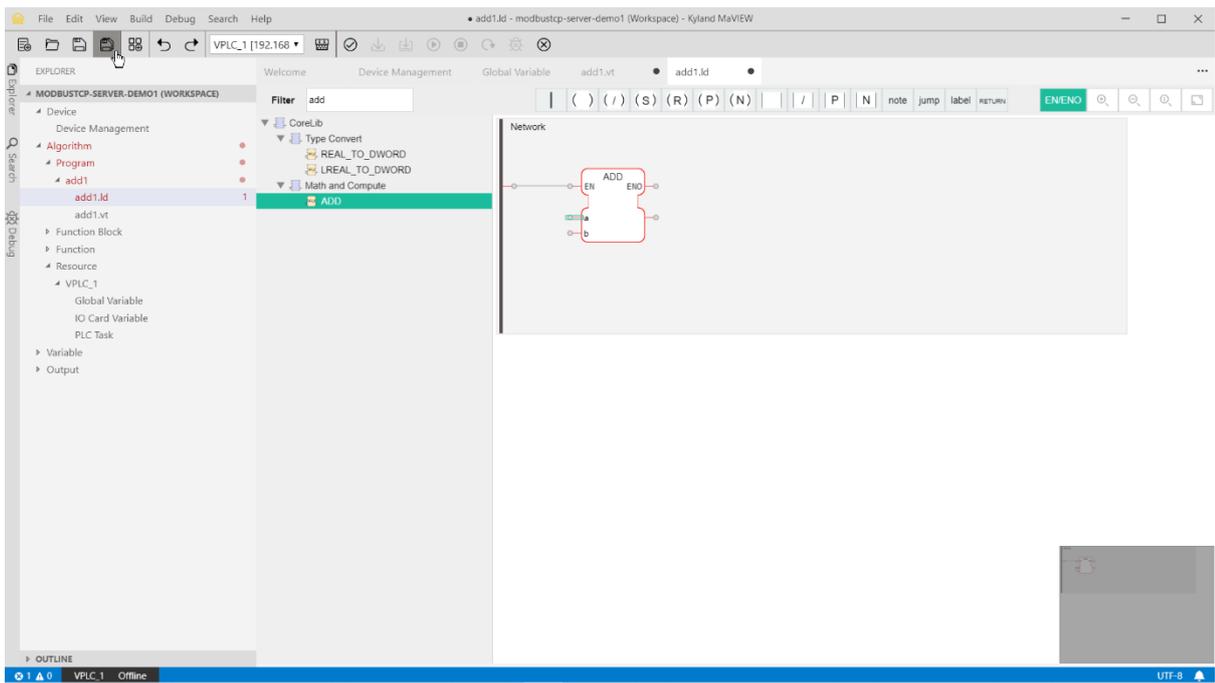
11. drag a 'PowerRail' into the working area

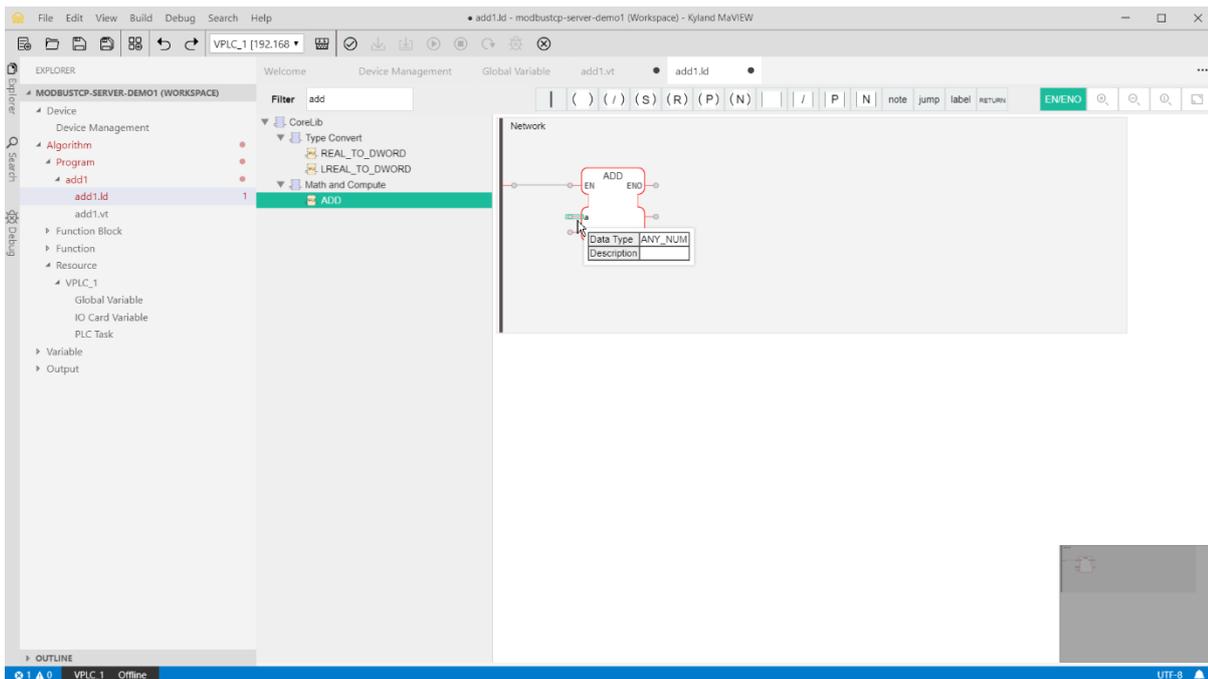


12. drag an 'ADD' operator next to the 'PowerRail'

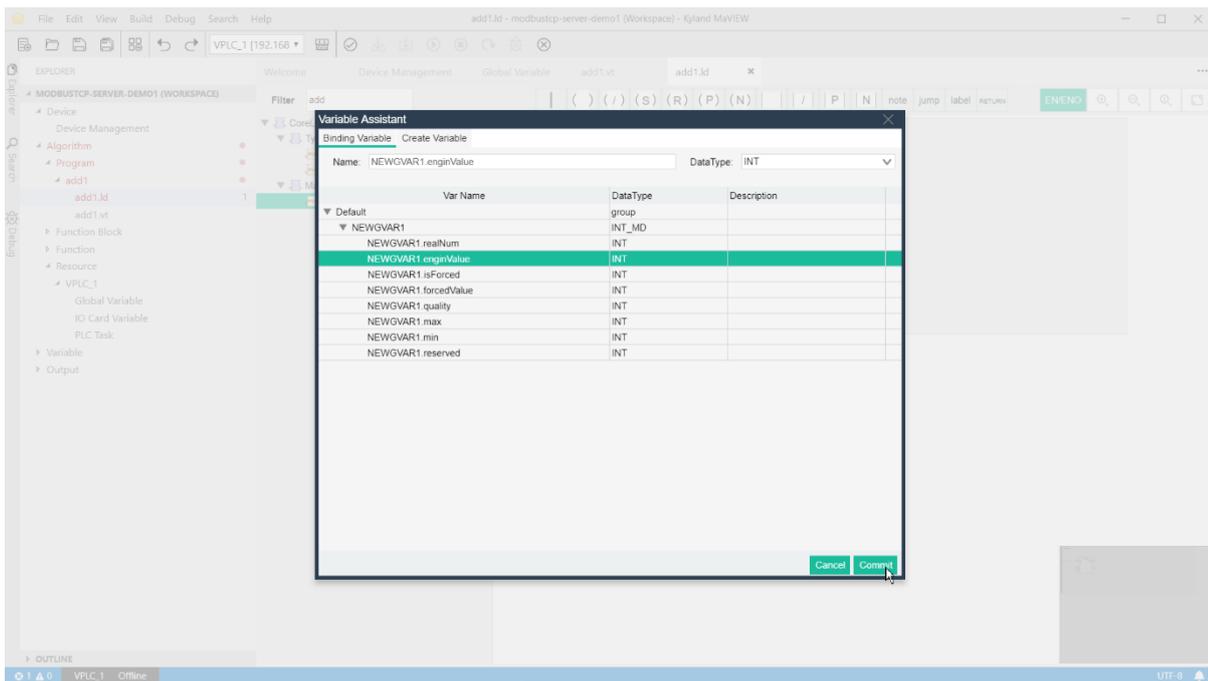


### 13. 'Save All' and double click 'a'

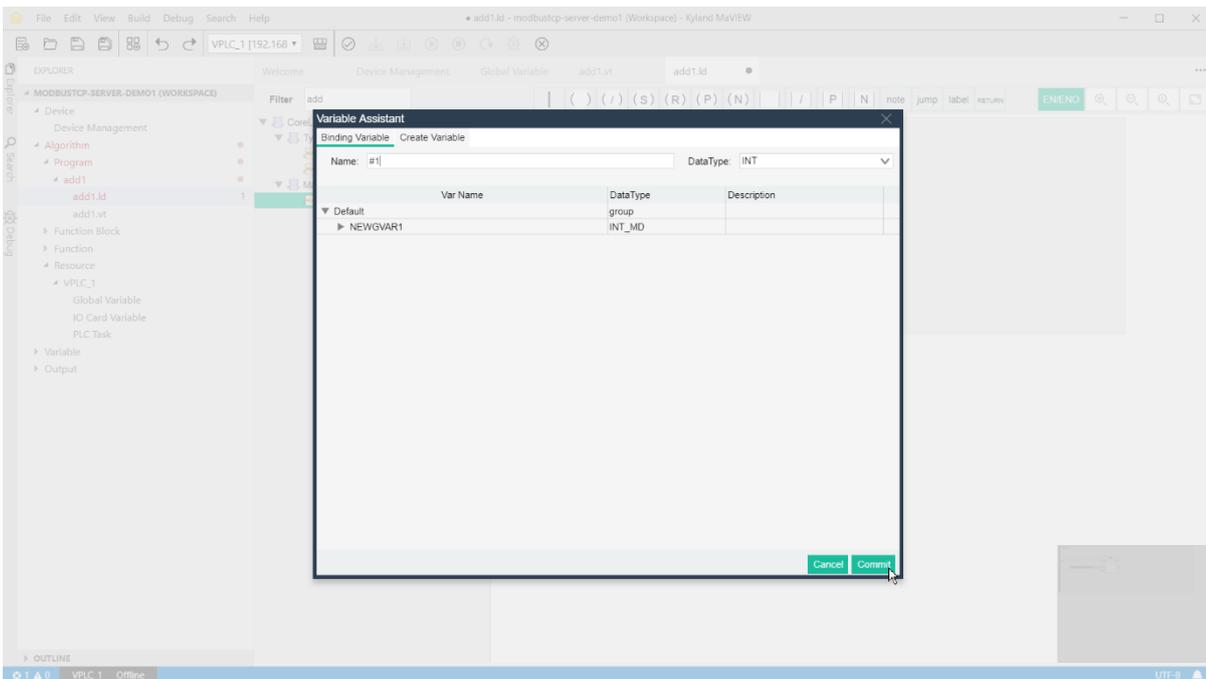
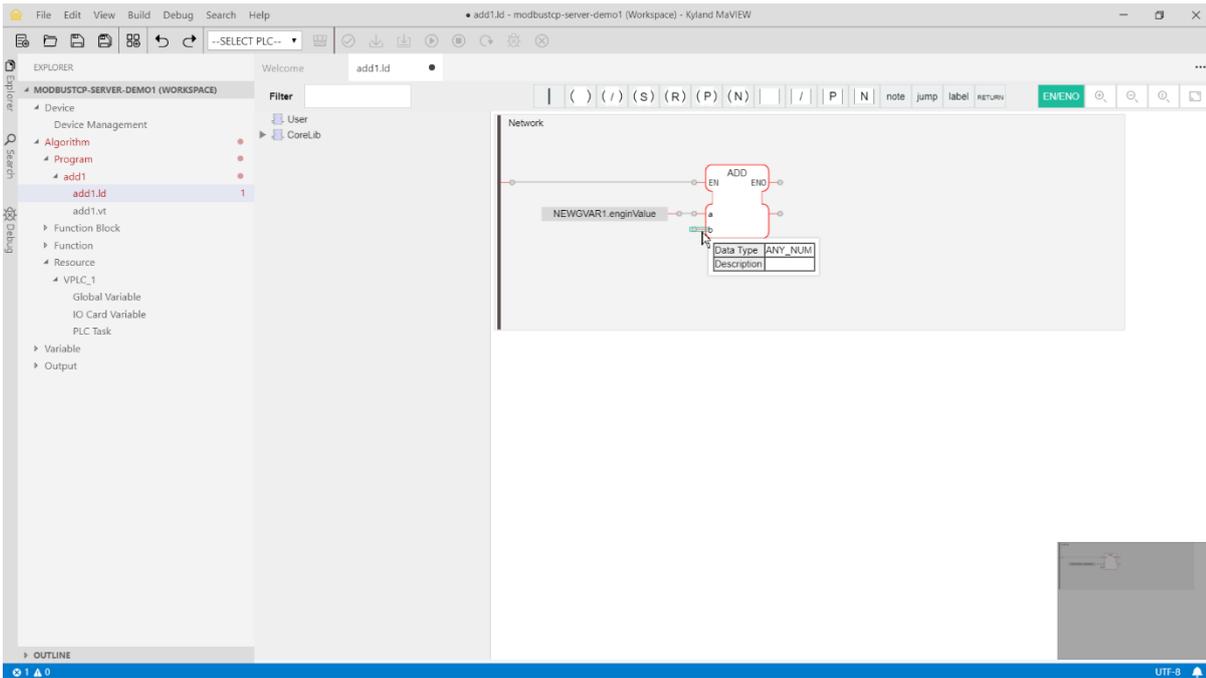




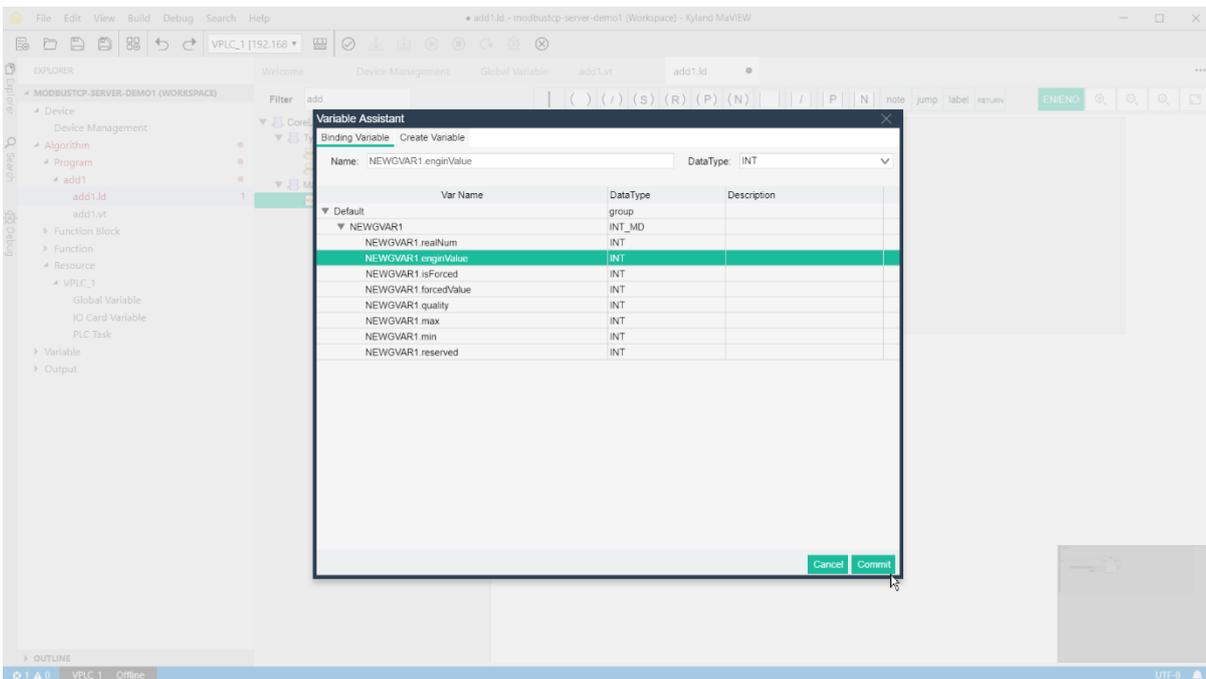
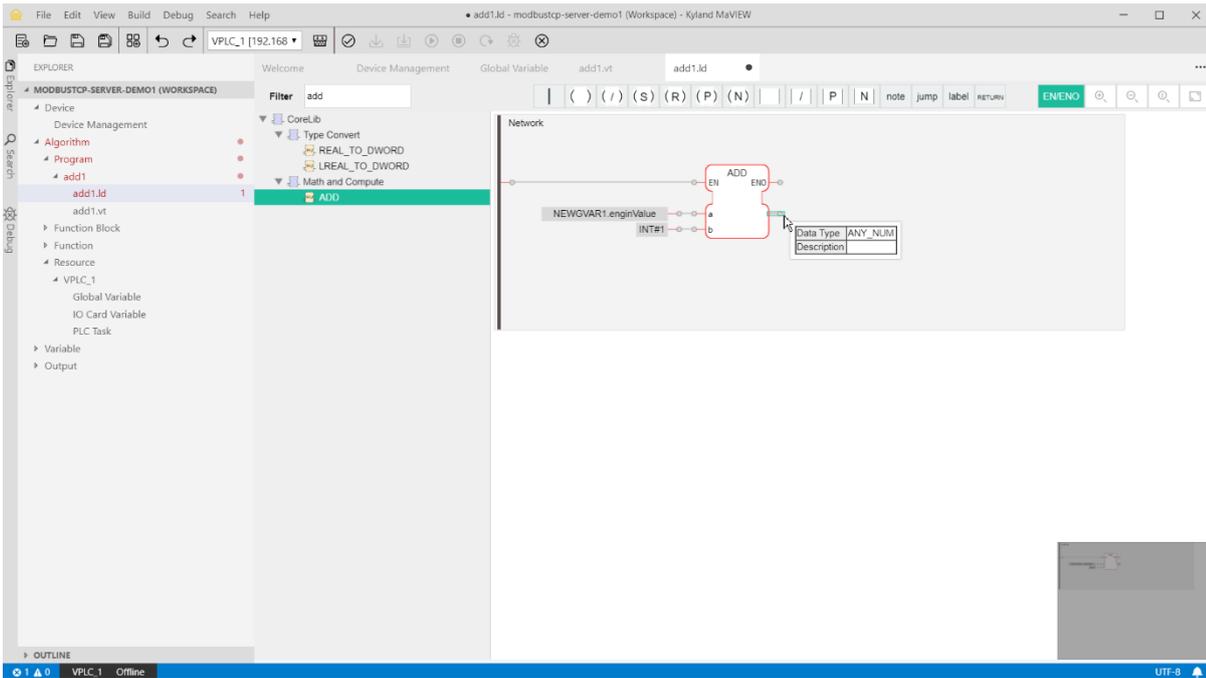
14. select 'NEWGVAR1.enginValue'



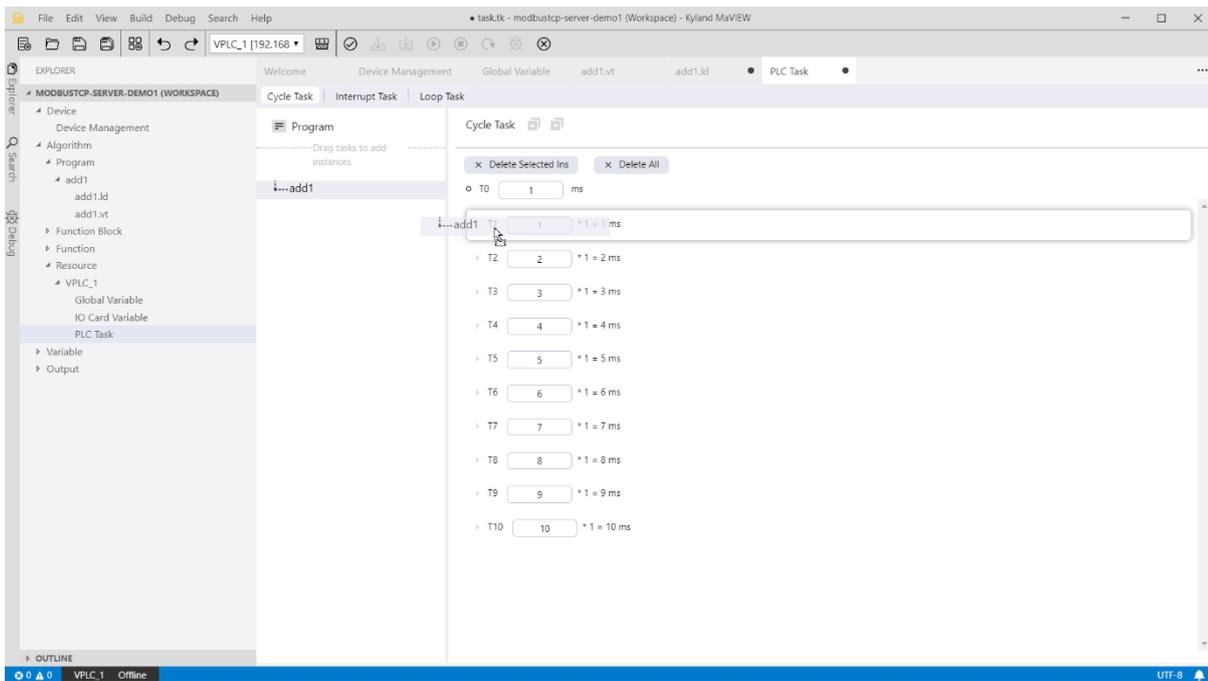
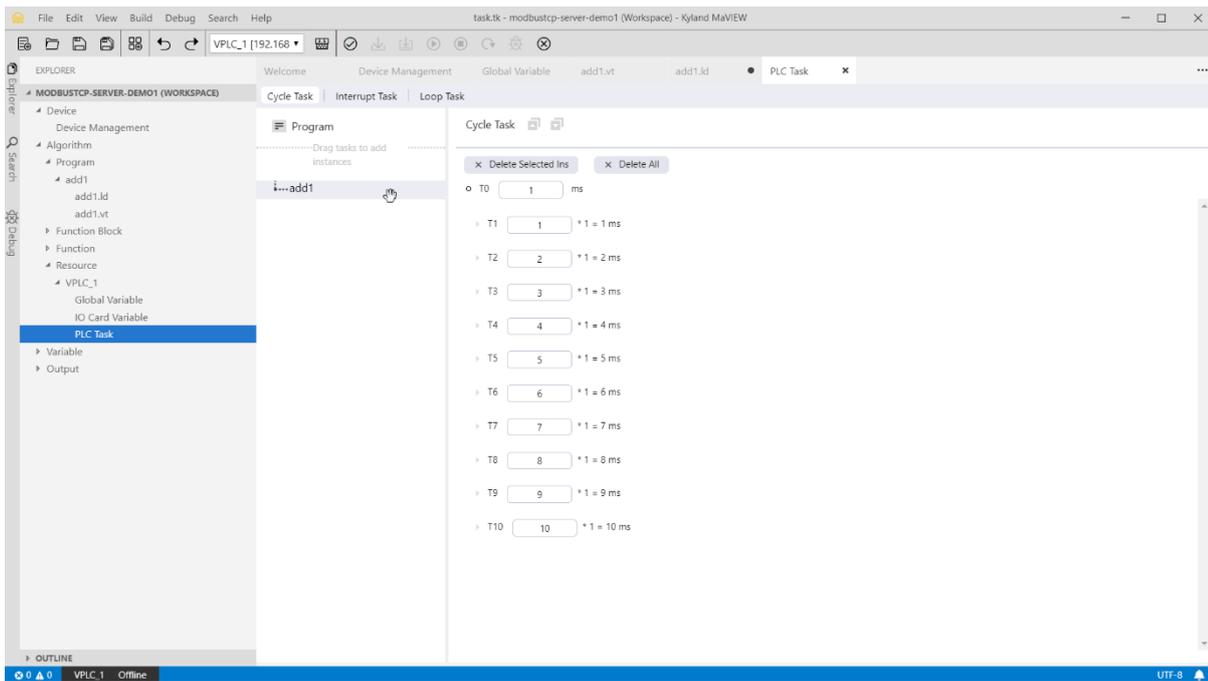
15. double click 'b', enter the name as '#1' and select Data Type 'INT'



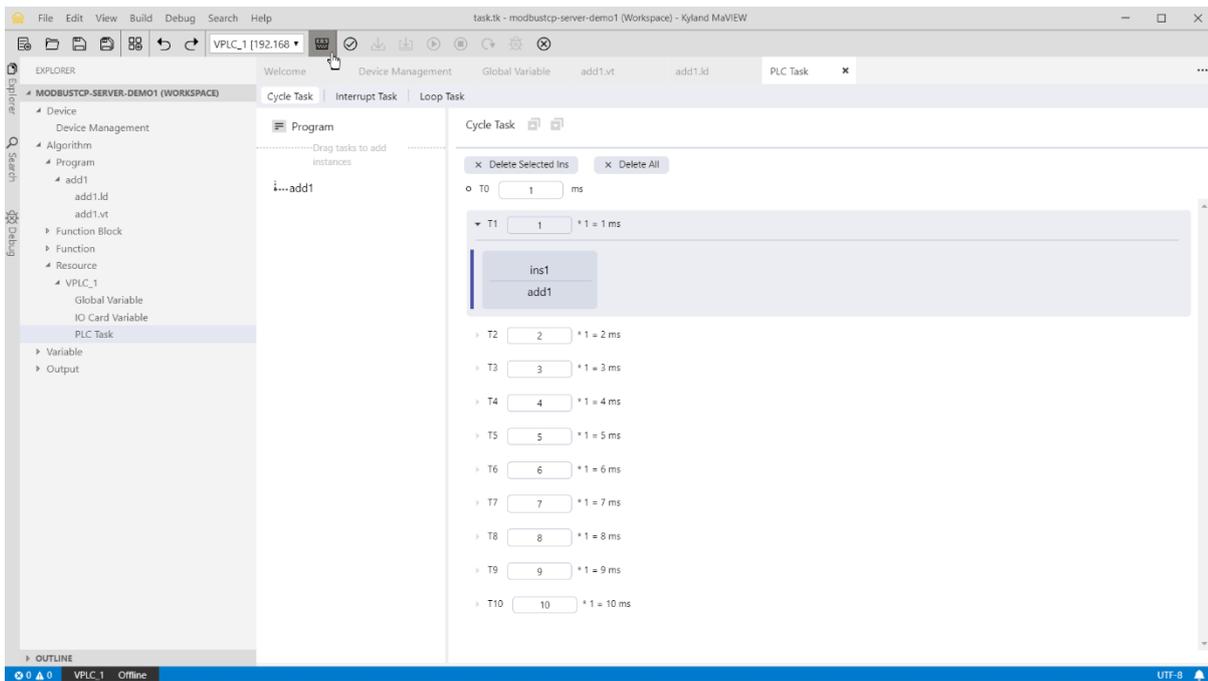
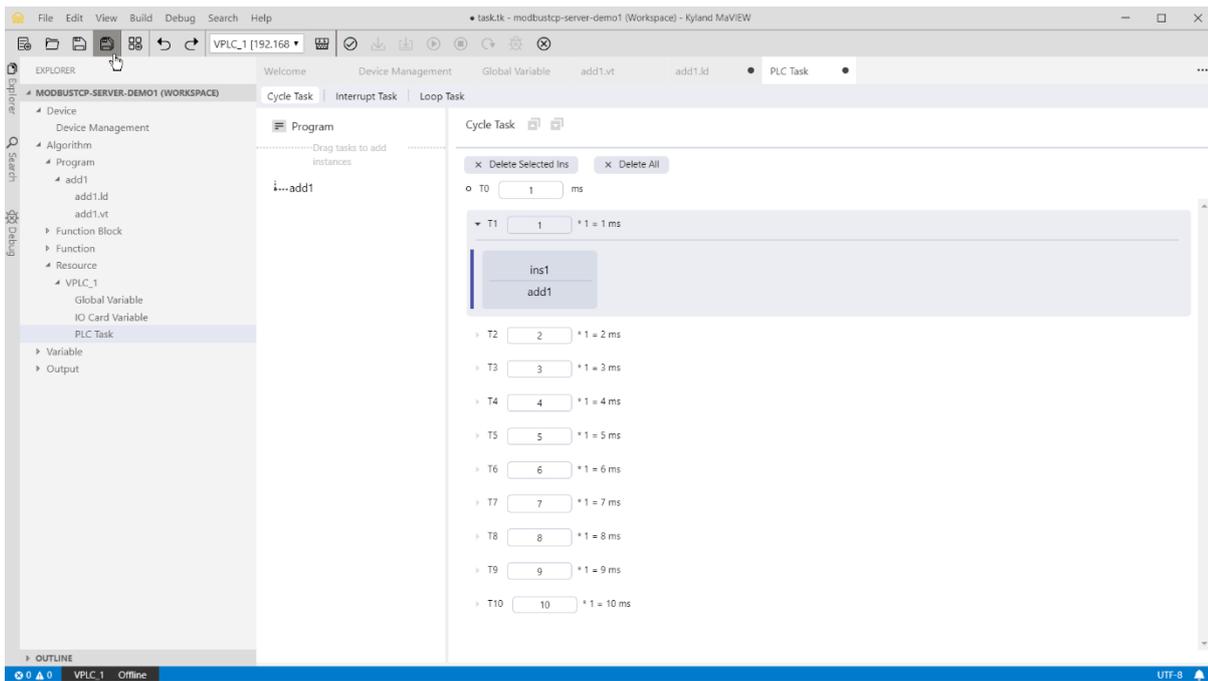
16. double click output and select 'NEWGVAR1.engineValue'



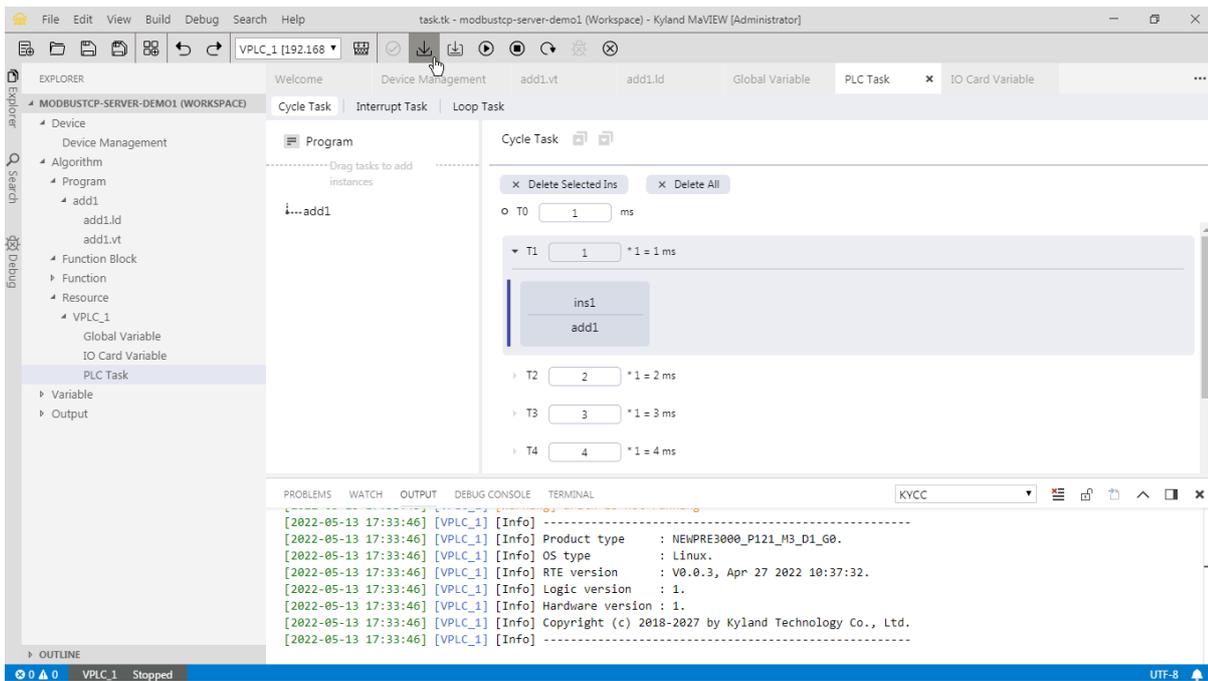
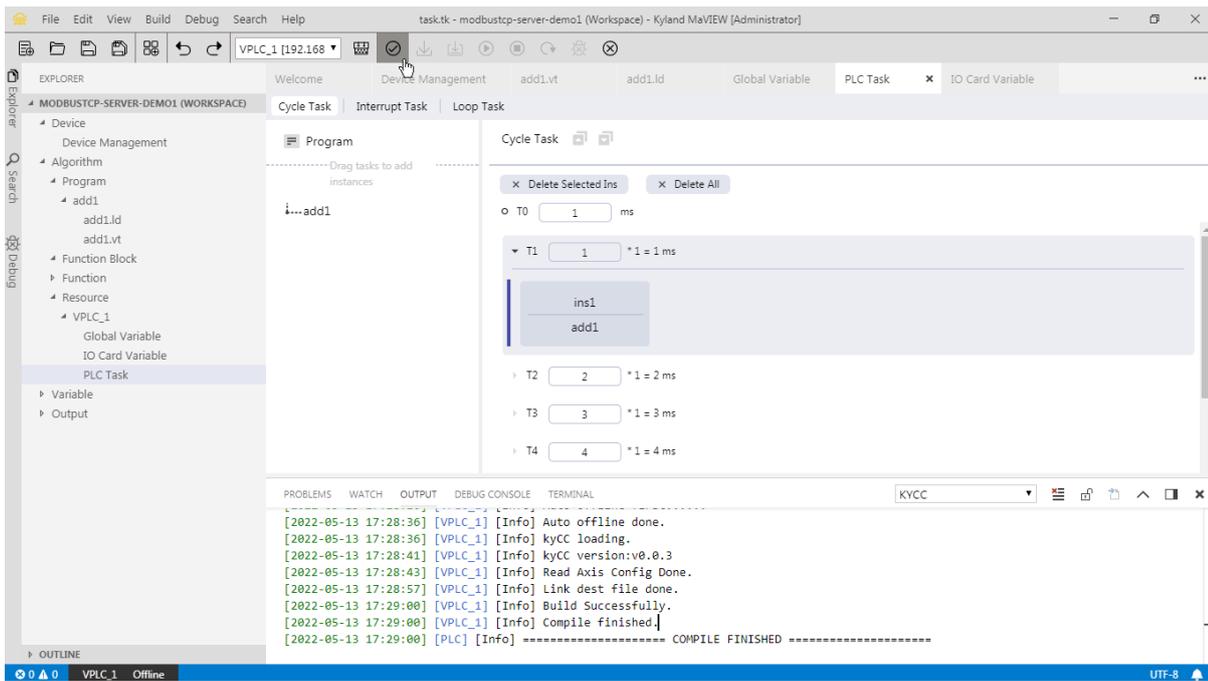
17. under the 'VPLC\_1', 'PLC' task, drag the above ladder program to the cycle task



## 18. 'Save all' and 'Compile PLC'

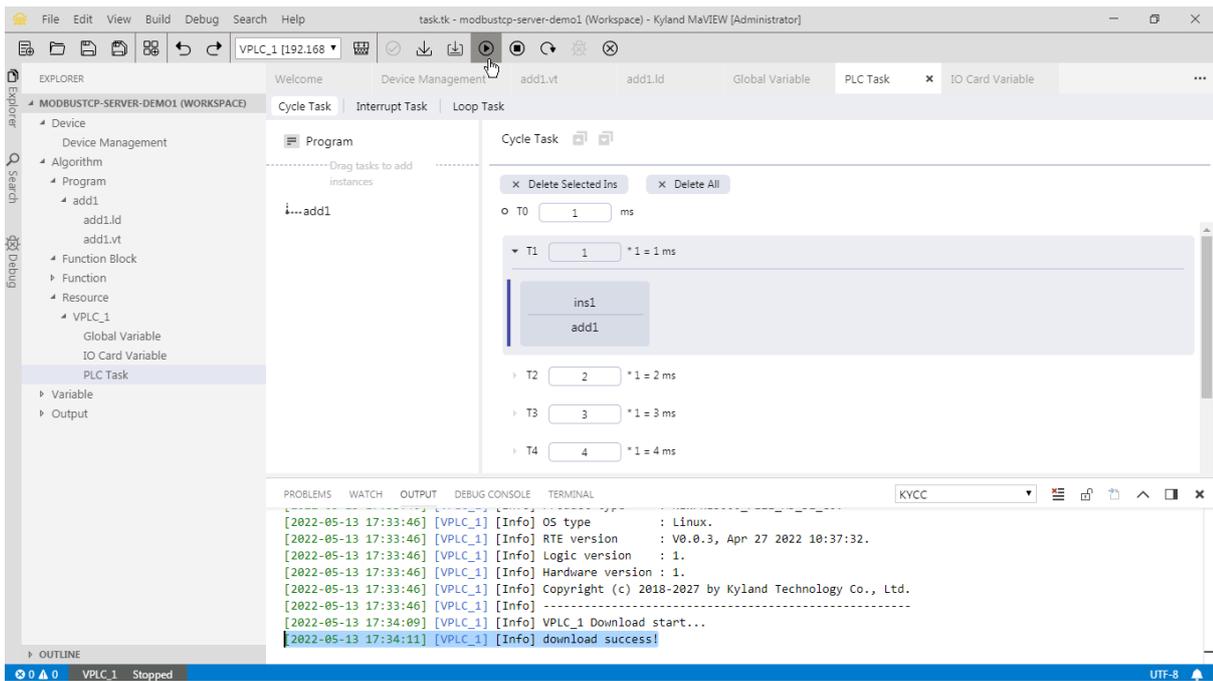


19. get 'Online' and then 'Coverage Download'



20. launch the sample program via 'Algorithm Start'

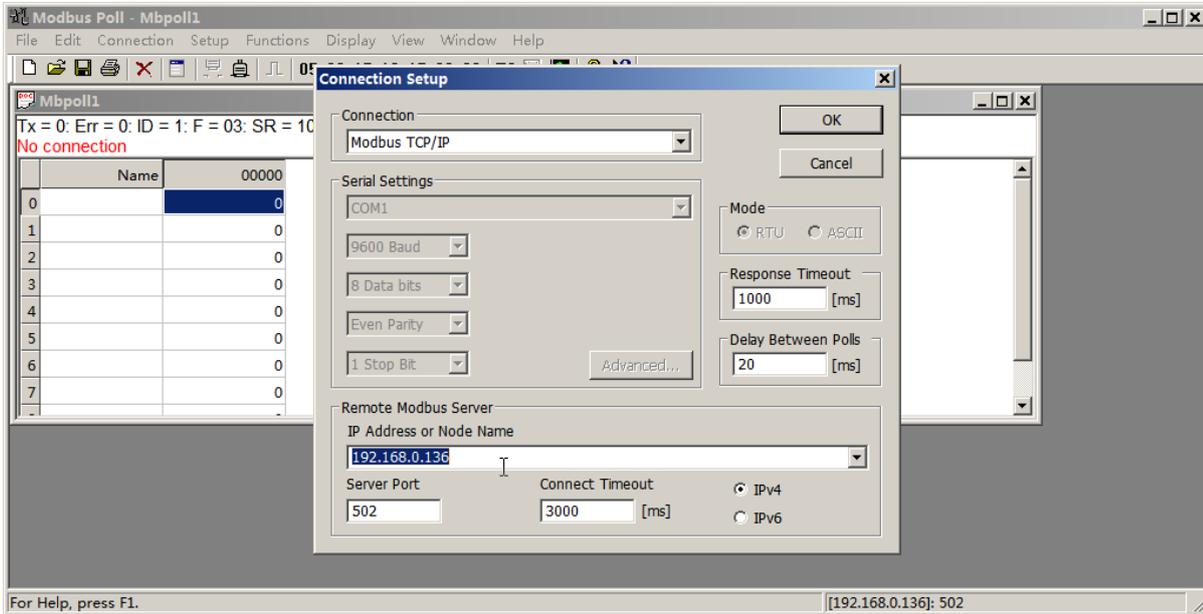
\* the 'program' will be downloaded and auto-executed at Target\_PC start-up



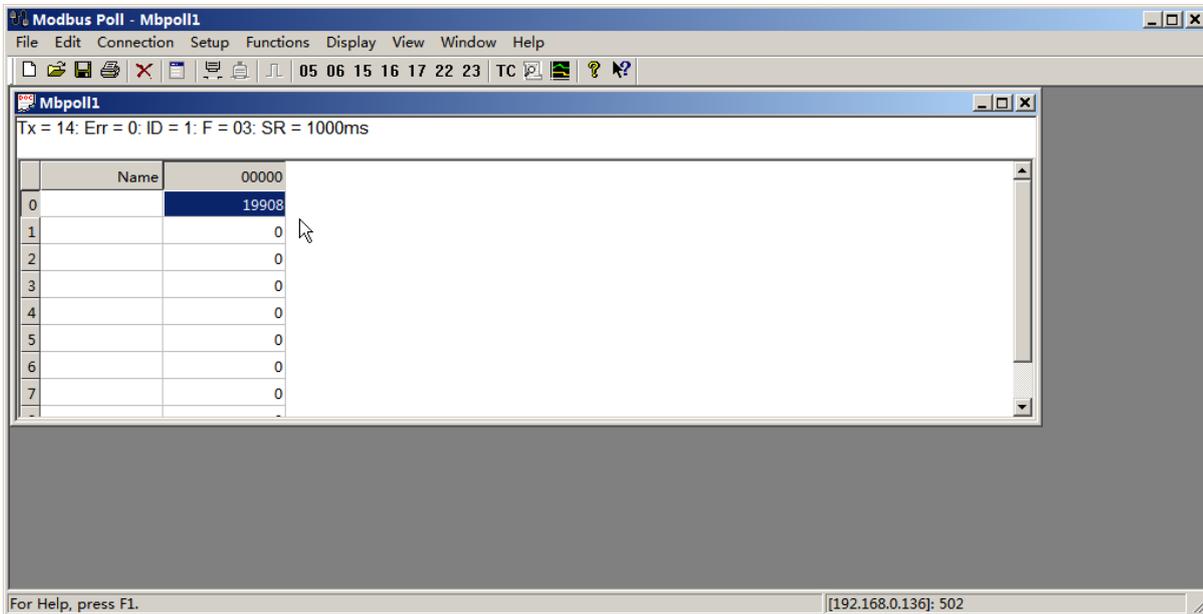
### C. Test the virtual PLC program (10mins)

\* use 'Modbus Poll' tool (free for trial) to test our Modbus-TCP Server sample program  
<https://www.modbustools.com/download.html>

1. setup the connection IP address as the **Target\_PC** (192.168.0.136)



2. poll the Modbus Slave Address of 400001 (address 0)



**D. Fill up the feedback form and attach your program (10mins)**

1. compress (zip) your project in the **Host\_PC**, and rename it as 'your\_name.zip'
2. fill up your feedback form and attach your project file:

## Appendix. Install and setup Ubuntu Linux (60-90mins)

1. download Ubuntu Linux ISO image

For example, an earlier version (smaller and stabler) of 16.04.7 server can be downloaded here:

<https://releases.ubuntu.com/xenial/>

<https://releases.ubuntu.com/xenial/ubuntu-16.04.7-server-amd64.iso> (880MB, without GUI)

<https://releases.ubuntu.com/xenial/ubuntu-16.04.7-desktop-amd64.iso> (1.6GB, with GUI)

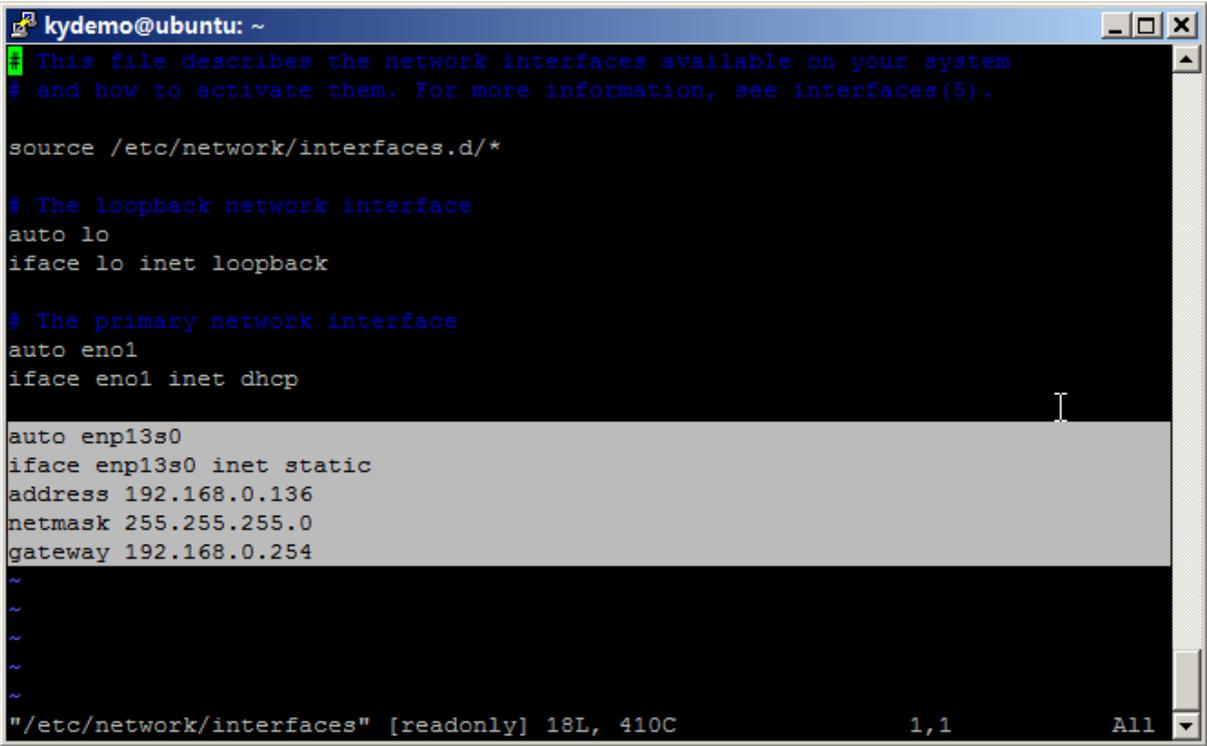
2. create a bootable USB drive via Rufus (freeware):

<https://ubuntu.com/tutorials/create-a-usb-stick-on-windows#1-overview>

3. boot the **Target\_PC** via the USB drive and follow the installation instruction

4. edit the /etc/network/interfaces to change the **Target\_PC** IP address permanently at start-up

kydemo@ubuntu:~\$ sudo vi /etc/network/interfaces



```
kydemo@ubuntu: ~
This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enol
iface enol inet dhcp

auto enp13s0
iface enp13s0 inet static
address 192.168.0.136
netmask 255.255.255.0
gateway 192.168.0.254
~
~
~
~
"/etc/network/interfaces" [readonly] 18L, 410C 1,1 All
```

\* 'enp13s0' is the Ethernet interface of the **Target\_PC** connecting to **Host\_PC**

\* under vi, type 'o' to add a new line, press 'Esc' and then type ':wq' to save the file (reboot to make it effective)