FAQ COMBIVIS

Ethercat gateway on Beckhoff   FAQ No.0006

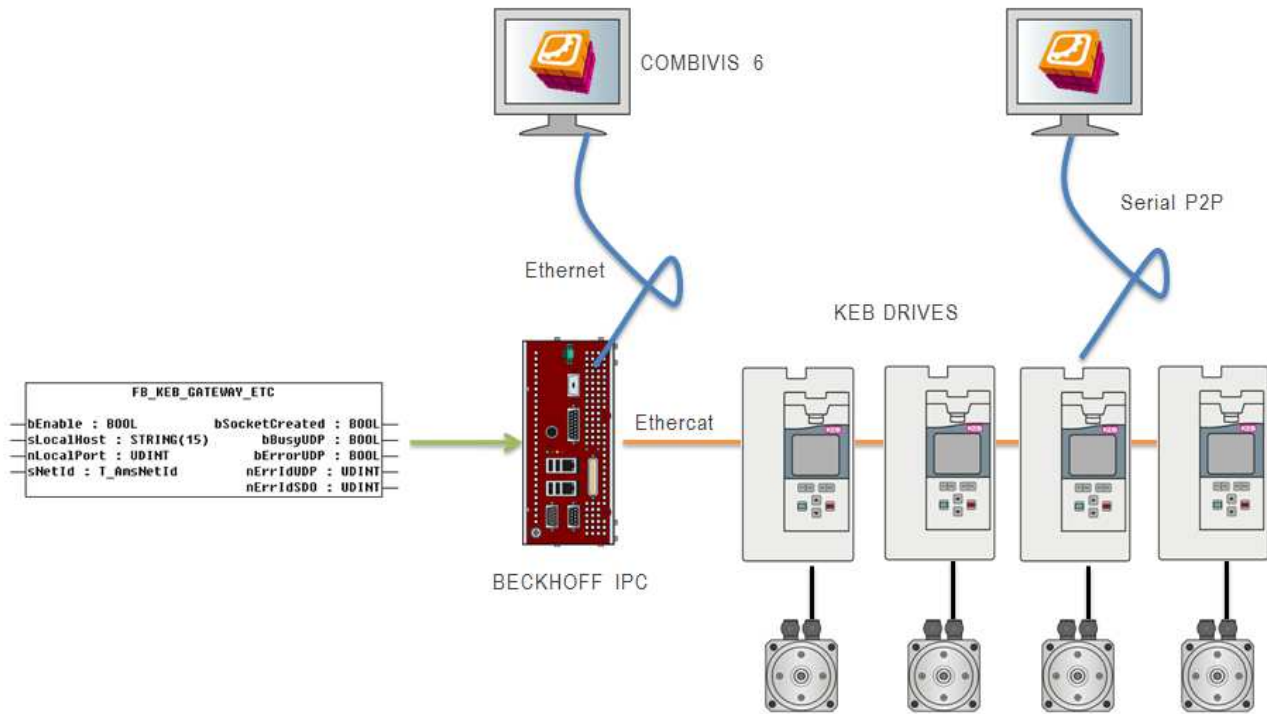| Part	Version	Revision	Date	Status
en	6.2.3.0	001	2019-01-01	Released

Content

Introduction

The KEB_Gateway_Din_Etc.lib is a tool helping startup the KEB devices on a Beckhoff PLC. It connects COMBIVIS through the EtherCAT bus directly to the drives. This way it's possible to startup several drives without connecting to every single drive directly.
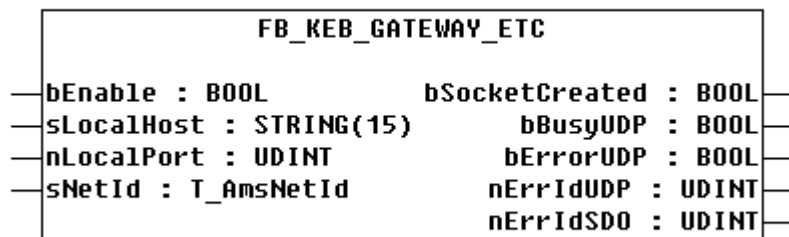
The function block accesses the UDP port of the PLC. Beckhoff offers a library (the TCPIP.lib) for socket communication on this port. As the KEB gateway is accessing the UDP port this lib is mandatory for the use of the Beckhoff gateway. You can download this library on the Beckhoff homepage in "Download/software/twincat 2 supplement/communication". The library needs a licence code. For a test version of this library just enter DEMO as licence code.

Implementation

```
                   FB_KEB_GATEWAY_ETC

—bEnable : BOOL            bSocketCreated : BOOL—
—sLocalHost : STRING(15)        bBusyUDP : BOOL—
—nLocalPort : UDINT            bErrorUDP : BOOL—
—sNetId : T_AmsNetId          nErrIdUDP : UDINT—
                             nErrIdSDO : UDINT—
```

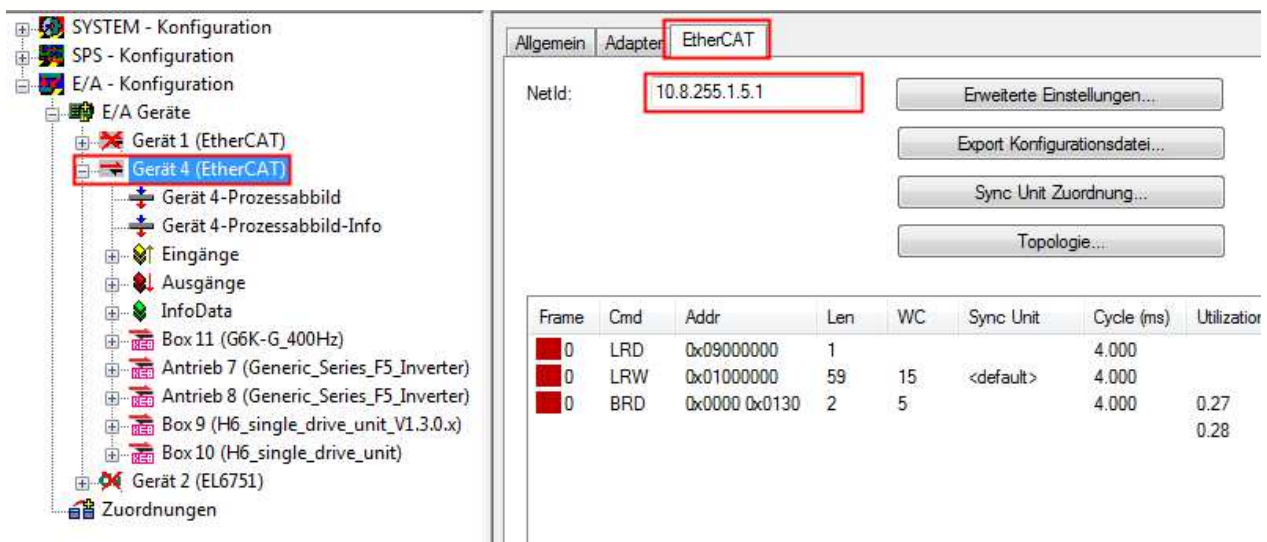Call the function block anywhere in your program (cyclic for good performance).

Inputs:

bEnable: This input starts up all the communication in the function block. It can also be used to shut down the communication and close the UDP socket.

sLocalHost: In this input the IP address of the Ethernet card which is connected to the COMBIVIS computer is entered. Some PLC's might have several network cards for different uses.

nLocalPort: In default COMBIVIS is always using the port 8000. This is also the default value for this input. So you only need to write to this input in case you want to use a different port.

sNetId: In this input you need to enter the AmsNetId of your EtherCAT master. To find the NetId check the settings of your EtherCAT master in your systemconfigurator. (the AmsNetId can be entered like a string)

Outputs:

bSocketCreated: This output shows if the socked was successfully created

bBusyUDP: This output shows if the socked is busy

bErrorUDP: This output shows if the UDP communication is corrupted in any kind of way

nErrIdUDP: This output forward the error ID created from the socket library from Beckhoff. You can find a documentation of the errors in the Beckhoff infosys "TwinCAT 2/TwinCAT Supplement/Kategorie Communication TS6xxx/TS6310|TwinCAT TCP/IP Connection Server/Anhang/fehlercodes" http://infosys.beckhoff.com/index.php?content=../content/1031/tcpipserver/html/tcplclibtcpip_overview.htm&id=

nErrIdSDO: This output forward the error ID (ADS error code) created from the Beckhoff SDO function block http://infosys.beckhoff.com/espanol.php?content=../content/1034/tcadsnetref/html/twincat.ads.adserrorcode.html&id=

Global Constants

In the global variables of the lib you find the MAX_ETC_SLAVES entry.
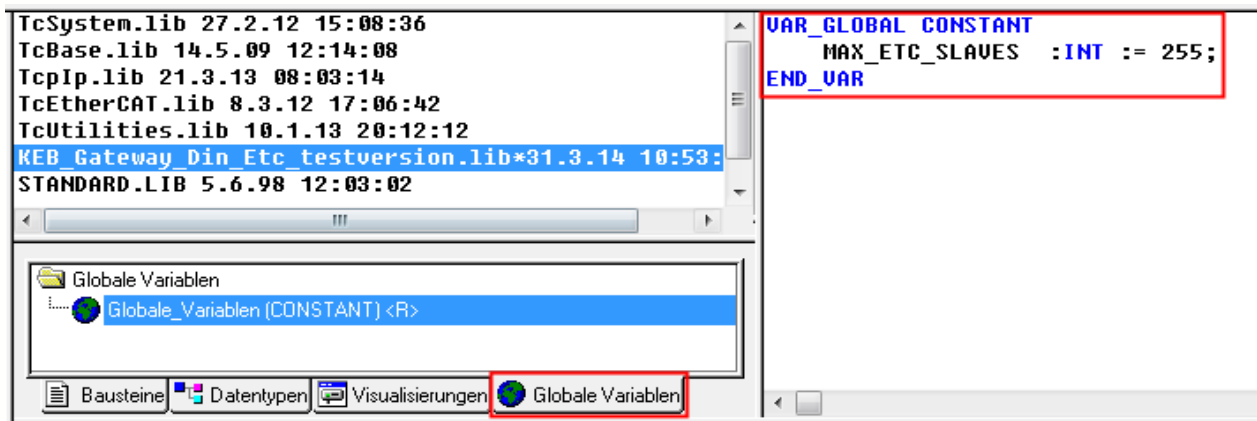
This value defines how many EtherCAT slaves are inside your EtherCAT master (overall). In default it's based on 255 slaves. If your network is smaller it's not a problem (except you really need to look at used memory in your application).

If you need more than 255 slaves, create a constant variable with the same name in your project and enter a higher number. This way the library constant gets replaced by your application constant.

How the process works

As soon as the enable is true, the FB scans the EtherCAT for KEB drives. After a successful scan they get stored as DIN66019 node addresses. The node address is pending on the position in the EtherCAT bus. It starts from node 0 and count up for every founded KEB drive.

e.g:	EtherCAT config:	KEB node address:
Etc addr 1001:	IO card	no KEB device
Etc addr 1002:	KEB F5	Node 0
Etc addr 1003:	Etc encoder	no KEB device
Etc addr 1004:	KEB G6	Node 1

Meanwhile the UDP socket gets created and starts recording requests from COMBIVIS. Every request gets the request source ID address assigned. This way it's also possible to access the gateway with several COMBIVIS users from different PC's.

As soon as the first valid UDP strings arrived the function block converts the DIN66019 requests to regular SDO read or write commands. Afterwards they get translated back to DIN66019 UDP strings and send to the request source.

Instance: System configuration

KEB device scan:

COMBIVIS:

Limitations:

COMBIVIS supports several services. Not all services are supported by the function block as the SDO communication does not support all these features.
For X5 generation devices:

- Active set is not supported

- Fast scope is not supported

- Multiple set selection is not supported

For X6 generation devices:

- Active set is not supported

- indirect set is not supported

- Multiple Set selection is not supported (there are no more sets in the X6 generation)

- Fast scope is not supported

You can switch between the different addressing modes by using the expert addressing settings. The CANopen compatible service works for all devices so this is our recommendation. It is also the most used service if you keep the settings on default so you probably don't need to touch this settings anyway.
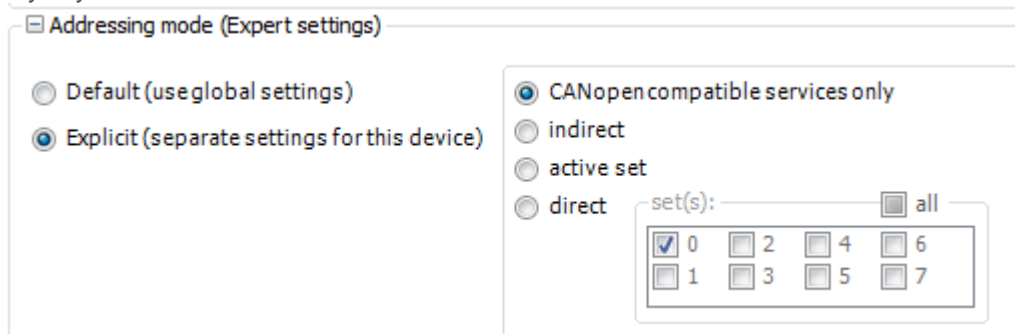
Hints:

In some cases Beckhoff devices have the **Windows firewall** active.
Switch off the firewall or add the port 8000 as exception to the settings to get the FB running.

If there is a lot of SDO traffic in your project in parallel to the FB, the respond time of the FB can be very slow. In such cases you can increase the COMBIVIS timeout time. [tools/options…/KEB Parameterization/Communication/Timeout [ms]]

Disclaimer

KEB Automation KG reserves the right to change/adapt specifications and technical data without prior notification. The safety and warning reference specified in this manual is not exhaustive. Although the manual and the information contained in it is made with care, KEB does not accept responsibility for misprint or other errors or resulting damages. The marks and product names are trademarks or registered trademarks of the respective title owners.

The information contained in the technical documentation, as well as any user-specific advice in verbal or in written form are made to the best of our knowledge and information about the application. However, they are considered for information only without responsibility. This also applies to any violation of industrial property rights of a third-party.

Inspection of our units in view of their suitability for the intended use must be done generally by the user. Inspections are particular necessary, if changes are executed, which serve for the further development or adaption of our products to the applications (hardware, software or download lists). Inspections must be repeated completely, even if only parts of hardware, software or download lists are modified.

Application and use of our units in the target products is outside of our control and therefore lies exclusively in the area of responsibility of the user.