FAQ C6 EMBEDDED

C6 Compact II Ethernet Powerlink Slave

FAQ No.0003

Part	Version	Revision	Date	Status
en	6.4.0	001	2019-01-01	Released

Content

KEB Automation KG
Südstraße 38
32683 Barntrup, Germany
E-mail: info@keb.de
Tel. +49 5263 401-0
Fax +49 5263 401-116
www.keb.de

District Court Lemgo HRA 5649
DUNS-No. 314108728
VAT-No. DE309087075
Bank Details: Sparkasse Paderborn-Detmold
IBAN DE 19 4765 0130 0000 0060 07
BIC WELADE3L

General Partner:
Vittorio Tavella
KEB Verwaltungs-GmbH, Barntrup
District Court: Lemgo HRB 8965
Directors: Curt Bauer CMO,
Ralf Lutter COO, Vittorio Tavella CFO,
Wolfgang Wiele CTO

Introduction

This document describes the usage of the C6 Compact II with the Ethernet POWERLINK (EPL) technology of the Ethernet POWERLINK Standardization Group (EPSG).
The C6 Compact II is working as slave (CN – Controlled Node) and needs an XML Device Description (.xdd) file for the network master (MN – Managed Node).

Because Ethernet POWERLINK is using the application layer of CANopen, it is also called “CANopen over Ethernet”[1].

.xdd – XML device description file

The XDD file contains an EPL device profile for the C6 Compact II identification and an EPL communication profile with the CANopen object dictionary.

For using COMBIVIS studio 6 applications with a default PDO mapping in the XDD file, only the EPL communication profile needs to be customized by the user’s needs. To support this task, the KEB Automation KG is providing a sample XDD file.

.xdd file validation

For a file validation check of modifications, the EPSG is providing[2] the openCONFORMANCE tool and a web based description file check:

openCONFORMANCE tool	: download for EPSG member only
Description file check	: web based, free of charge for all registered user

Device identification

The C6 Compact II will be identified at least by the values:

Vendor ID	: 00000014h	(KEB Automation KG)
Product ID	: 500100	(C6 Compact II with Ethernet POWERLINK)

Communication object mapping

In the EPL communication profile each object for the PDO mapping needs to be added as subobject to the PDO mapping objects 1600h (Rx) and 1A00h (Tx).

The subobjects (default) value contains the mapped object’s (sub-) index, its length and the offset to the beginning of the PDO mapping in the following order:

Length	(16 bit, MSB)
Offset	(16 bit)
Reserved	(8 bit)
Sub-index	(8 bit)
Index	(16 bit, LSB)

It is useful to enter the value in hexadecimal way.

[1] https://www.ethernet-powerlink.org/powerlink/technology/canopen-and-powerlink
[2] https://www.ethernet-powerlink.org/powerlink/conformity

Example PDO mapping
In this example, the user objects 4000h (Rx) and 4010h (Tx) will be mapped.

The first Rx user object, located in 4000:01h, is defined by the object type 7 (VAR[3]) and has the data type 6 (Unsigned16 / WORD, 16 bit).

For the first Rx PDO mapping of user objects in the Rx PDO 1600:01h, the mapping of the user object 4000:01h will be done with the following value:

Length : 0010h (16 bit length of the variable)
PDO offset : 0000h (first mapping, no length offset yet)
Reserved : 00h
Sub-index : 01h
Index : 4000h

In combination does this look like as followed: 0010000000014000h.

Every next Rx PDO mapping needs to have added the previous length to the previous length offset and to be added as next subobject of the Rx PDO 1600h.

The number of added and used subobject has to be equal to the value in object 1600:00 “number of entries”.

The Tx PDO mapping is working in the same way.

[3] EPSG 301 v1.1.0 DS Tab. 37 Object type definitions

COMBIVIS studio 6 sample project

In the COMBIVIS studio 6 project “Sample_C6C2_PowerLinkSlave.project” the program “RunPowerLink” provides the access to an Ethernet POWERLINK node of the KEB library KEB_RtEthernet and an input / output buffer of DWORDs.

RunPowerLink

```
1  PROGRAM RunPowerLink
2  VAR
3      PowerLinkNode : KEB_RtEthernet.EplNode;
4      ProcessDataIn : ARRAY[0..31] OF DWORD;
5      ProcessDataOut : ARRAY[0..31] OF DWORD;
6  END_VAR
7
```

The input / output buffers are connected to the .PdIn / .PdOut variables by its address and its maximum size.

```
14     //Set address of processdata buffers
15     PowerLinkNode.PdIn.pBuffer := ADR(ProcessDataIn);
16     PowerLinkNode.PdOut.pBuffer := ADR(ProcessDataOut);
17
18     //Set maximum size of processdata image
19     PowerLinkNode.PdIn.MaxBufLen := SIZEOF(ProcessDataIn);
20     PowerLinkNode.PdOut.MaxBufLen := SIZEOF(ProcessDataOut);
```

The total process data length is defined in the XDD file, by calculating of the number of mapped objects with their size in byte, and can be set to the ProjectedSize value in the .PdIn / .PdOut variables.

```
24     PowerLinkNode.PdIn.ProjectedSize := 8;
25     PowerLinkNode.PdOut.ProjectedSize := 8;
```

The network IP address settings are shared in a global variable list for every fieldbus, which the internal netX Ethernet controller is supporting. The IP address of the EPL node will be attached to the class C network address “192.168.100.x”[4] and has to fit the settings in the EPL master project.
The gateway address is set to “192.168.100.254” in default and is needed to communicate with hosts and services in foreign IP address ranges.

```
34     KEB_RtEthernet.GVL_NETX_RETAIN.DeviceId := 2;
35     KEB_RtEthernet.GVL_NETX_RETAIN.GwAddress := 16#C0A864FE;
```

[4] EPSG 301 v1.1.0 DS Tab. 29 IP parameters of a POWERLINK node

Sample data exchange
Some few instructions will be processed when the EPL node has been switching to the network state operational.

Here, any input of the array item ProcessDataIn[0] will be added to the DWORD 11111111h and returned as array item ProcessDataOut[0].

In the array item ProcessDataOut[1], a continuously updated actual time value will be send to the EPL master.

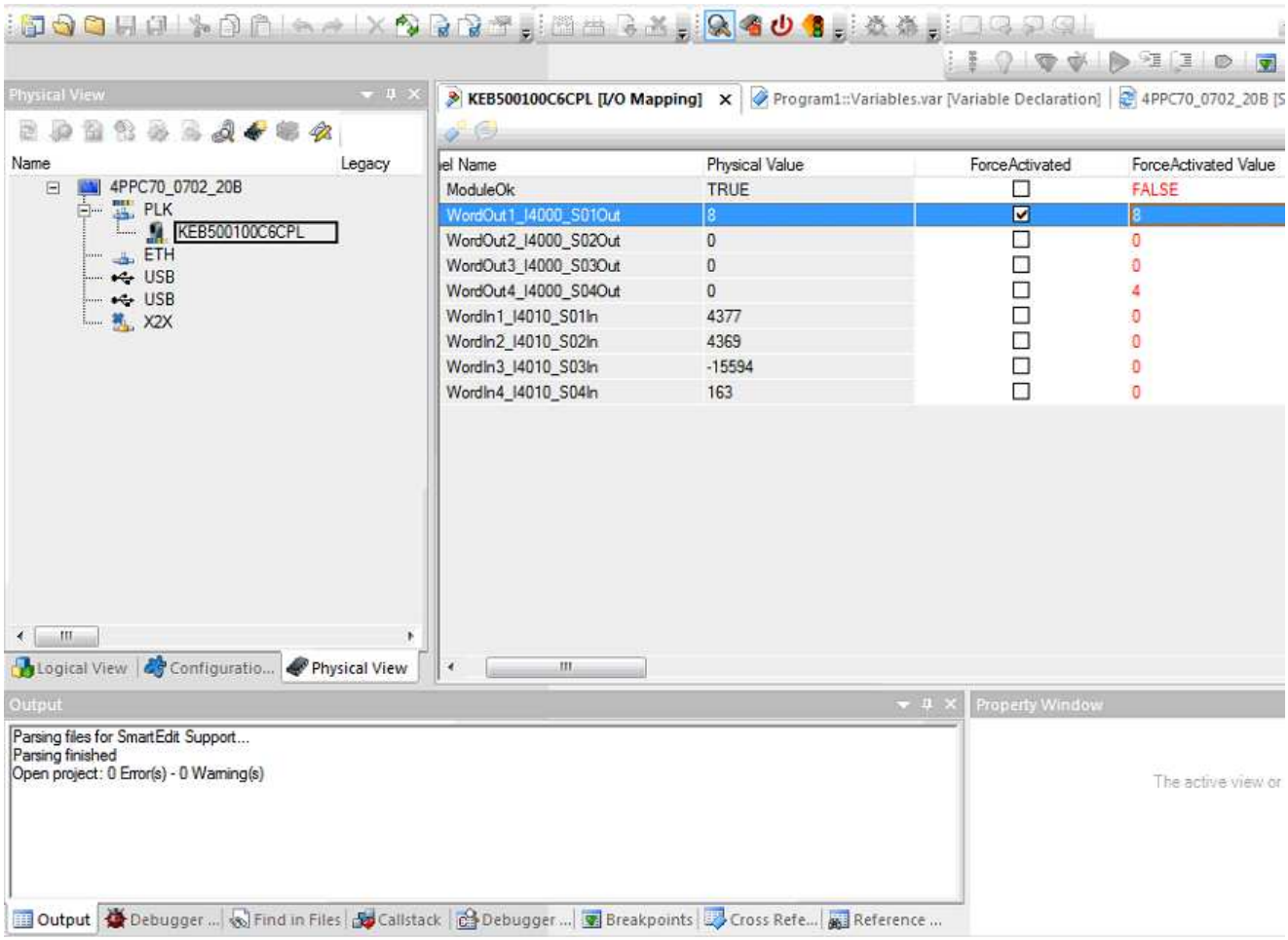
```
65  IF PowerLinkNode.NMT_State = KEB_RtEthernet.T_EPL_NMT_STATE.OPERATIONAL THEN
66      ProcessDataOut[0] := ProcessDataIn[0] + 16#11111111;
67      KEB_Base.DwordToBuffer(TIME_TO_DWORD(TIME()),ADR(ProcessDataOut[1]),TRUE);
68  END_IF
```

These data values will be shown in the next Automation Studio sample screenshot.

Automation Studio sample screenshots

The following screenshot shows a sample I/O mapping in an Automation Studio instance.

The EPL master transmits the value “8” by the WordOut1 and, with the adding instructions of the COMBIVIS studio 6 project, the result will be returned to the variables WordIn1 and WordIn2 (11111119h).
The time value will be received in the variables WordIn3 and WordIn4.



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.

KEB Automation KG
Südstraße 38 • D-32683 Barntrup
fon: +49 5263 401-0 • fax: +49 5263 401-116
net: www.keb.de • mail: info@keb.de